

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ  
СІКОРСЬКОГО»  
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ  
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

На правах рукопису  
УДК 519.688

До захисту допущено  
В. О. завідувача кафедри ММСА

О.Л.Тимошук  
«\_\_\_» \_\_\_\_\_ 2020 р.

## **Магістерська дисертація**

на здобуття ступеня магістра за спеціальністю 124 Системний аналіз  
на тему: «Система підтримки прийняття рішень для прогнозування рівнів  
продажу на підприємстві»

Виконав:

студент II курсу, групи КА-92мп  
Худецький Михайло Анатолійович

\_\_\_\_\_

Керівник:

професор кафедри ММСА,  
д.т.н, проф. Бідюк П.І.

\_\_\_\_\_

Рецензент:

декан ФІОТ КПІ ім. І. Сікорського  
д.т.н., проф. Теленик С.Ф.

\_\_\_\_\_

Засвідчую, що у цій магістерській  
дисертації немає запозичень з праць інших  
авторів без відповідних посилань

Студент \_\_\_\_\_

Київ  
2020

**Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Інститут прикладного системного аналізу  
Кафедра математичних методів системного аналізу**

Рівень вищої освіти – другий (магістерський)

Спеціальність (спеціалізація) – 124 «Системний аналіз» («Системний аналіз фінансового ринку»)

ЗАТВЕРДЖУЮ

Завідувач кафедри

\_\_\_\_\_ О.Л. Тимощук

«\_\_\_» \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ  
на магістерську дисертацію студенту  
Худецькому Михайлу Анатолійовичу**

1. **Тема дисертації** «Система підтримки прийняття рішень для прогнозування рівнів продажу на підприємстві», науковий керівник дисертації Бідюк Петро Іванович, доктор технічних наук, професор, затверджені наказом по університету від «02» листопада 2020 р. № 3182-с

2. **Термін подання студентом дисертації** 14 грудня 2020

3. **Об'єкт дослідження:** Статистичні дані про обсяг продажів мережі магазинів

4. **Предмет дослідження:** Система підтримки прийняття рішень для прогнозування рівнів продажу

5. **Перелік завдань, які потрібно розробити:** Проаналізувати предметну область, та на її основі запропонувати методи прогнозування. Використовуючи обраний метод, розробити систему підтримки прийняття рішень. Зробити висновок.

6. **Орієнтовний перелік графічного (ілюстративного) матеріалу:**

- 1) Таблиці
- 2) Блок схеми структури СППР
- 3) Зображення інтерфейсу СППР

7. **Дата видачі завдання:** 1 вересня 2019 року

### Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів роботи	Примітка
1	Формулювання тематики (напрямку) дослідження.	07.09.2019 – 30.09.2019	
2	Аналіз актуальності задач стосовно тематики дослідження	01.10.2019 – 30.10.2019	
3	Аналіз відомих результатів стосовно тематики дослідження	01.11.2019 – 30.11.2019	
4	Формулювання задач дослідження	01.12.2019 – 30.12.2019	
5	Уточнення (формулювання) теми дисертації	25.02.2020 – 28.02.2020	
6	Вибір методів виконання досліджень	01.03.2020 – 30.03.2020	
7	Виконання теоретичного дослідження	01.04.2020 – 30.05.2020	
8	Збір статичних даних, попередній аналіз даних, підготовка до моделювання	01.06.2020 – 30.08.2020	
9	Розробка (вибір) програмного продукту для виконання обчислювальних експериментів	01.09.2020 – 11.10.2020	
10	Виконання обчислювальних експериментів, аналіз та оформлення результатів	11.10.2020 – 10.11.2020	
11	Оформлення пояснювальної записки у цілому	11.11.2020 – 21.12.2020	
12	Захист дисертації	21.12.2020 – 23.12.2020	

Студент

М.А. Худецький

Науковий керівник дисертації

П.І.Бідюк

## РЕФЕРАТ

Магістерська дисертація: 82 с., 22 табл., 4 рис. 4 дод. 9 джерел

Об'єкт дослідження - статистичні дані про обсяг продажів мережі магазинів.

Предмет дослідження – система підтримки прийняття рішень для прогнозування рівнів продажу.

Мета роботи – розробити СППР для аналізу та прогнозування рівнів продажу на підприємстві.

Метод дослідження – побудова прогнозних моделей.

Актуальність роботи - створення простого у використанні програмного продукту, що дозволить робити прогнози рівнів продажів підприємства.

Програмний продукт реалізовано за допомогою мови програмування C#, на базі платформи .Net Core 5. Користувацький інтерфейс розроблено мовою JavaScript за допомогою бібліотеки ReactJS. Нейронні мережі побудовані за допомогою мови Python.

Отримані результати – розроблено систему підтримки прийняття рішень для прогнозування рівнів продажів на підприємстві, використовуючи модель багатошарового перцептрону.

НЕЙРОННІ МЕРЕЖІ, ПЕРЦЕПТРОН, БАГАТОШАРОВИЙ ПЕРЦЕПТРОН, РАДІАЛЬНО БАЗИСНІ ФУНКЦІЇ, ЧАСОВІ РЯДИ, МОДЕЛІ, ПРОДАЖІ, ПРОГНОЗУВАННЯ, .NET CORE, JAVASCRIPT, PYTHON.

## **ABSTRACT**

Theme: “DSS for enterprise sales forecasting”

Thesis: 82pages, 22 tables, 4 picture, 4 additions, 9 cited sources.

Object of the study - statistical data on sales revenue of store chain.

Subject of research – DSS for sales forecasting.

The purpose of the work is to develop a DSS for analysis and forecasting of enterprise sales volume.

The method of research – development of forecasting models.

Actuality is to work is to create an easy-to-use software product that will allow you to make forecasts of the company's sales volumes.

Application is implemented using the C # programming language, based on the .Net Core 5 framework. User interface was developed in JavaScript using the ReactJS library. Neural networks are built using Python.

Obtained results - the information-analytical system for the modeling and forecasting of weather processes using the autoregressive models and the least squares method is developed.

NEURAL NETWORKS, PERCEPTRON, MULTILAYER PERCEPTRON,  
RADIAL BASIS FUNCTIONS, TIME SERIES, MODELS, SALES,  
FORECASTING, .NET CORE, JAVASCRIPT,PYTHON

## Зміст

ВСТУП .....	9
1 ОСОБЛИВОСТІ ПРОГНОЗУВАННЯ ПРОДАЖІВ ПІДПРИЄМСТВА .....	10
1.1 Вплив прогнозу продажів, та фактори що на нього впливають .....	10
1.2 Методи прогнозування продажів .....	12
1.2.1 Якісні методи прогнозування продажів .....	14
1.2.2 Кількісні методи прогнозування продажів.....	17
Висновки до розділу .....	20
2 ВИБІР МАТЕМАТИЧНИХ МЕТОДІВ ДЛЯ ПОБУДОВИ МОДЕЛЕЙ ТА ПРОГНОЗУВАННЯ ПРОДАЖІВ.....	22
2.1 Моделі що використовуються для опису та прогнозування часових рядів .....	22
2.1.1 Модель авторегресії.....	22
2.1.2 Модель ковзного середнього .....	23
2.1.3 Модель авторегресії з ковзним середнім (АРКС) .....	24
2.2 Методика побудови моделей нестационарних процесів .....	25
2.2.1 Аналіз процесу та попередня обробка даних.....	25
2.2.2 Відбір структур моделі .....	26
2.2.3 Оцінювання параметрів моделей .....	28
2.3 Критерії адекватності моделей .....	29
2.4 Нейронні мережі що використовуються для прогнозування часових рядів .....	31
2.4.1 Багатошаровий перцептрон .....	33
2.4.2 Метод зворотнього поширення похибки.....	36
2.4.3 Нейронні мережі РБФ.....	40

Висновки до розділу .....	44
3 ПОБУДОВА МОДЕЛЕЙ ТА РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ ..	45
3.1 Огляд, аналіз та попередня обробка використаних даних.....	45
3.2 Побудова та оцінка моделей .....	46
3.2.1 Моделі авторегресії.....	46
3.2.2 Моделі навчені нейронними мережами.....	48
3.3 Загальний огляд реалізованої СППР .....	51
3.4 Огляд інтерфейсу користувача .....	53
Висновки до розділу .....	58
4 СТАРТАП АНАЛІЗ ПРОЕКТУ .....	60
4.1 Інформаційна карта проекту .....	60
4.2 Технологічний аудит ідеї проекту.....	62
4.3 Аналіз ринкових можливостей запуску стартап-проекту.....	65
4.4 Розроблення ринкової стратегії проекту .....	72
4.5 Розроблення маркетингової програми стартап-проекту .....	75
4.6 Висновки до розділу .....	76
ВИСНОВКИ.....	77
ПЕРЕЛІК ПОСИЛАНЬ.....	78
ДОДАТОК А.....	79
ДОДАТОК Б .....	81

## ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

СППР – системи підтримки прийняття рішень

НМ – нейронна мережа

ПП – програмний продукт

АРКС – авторегресія з ковзним середнім

АКФ – автокореляційна функція

ЧАКФ – часткова автокореляційна функція

РБФ – радіальна базисна функція



## ВСТУП

З ростом інформаційних технологій, з кожним роком об'єми інформації, що передається та повинна бути оброблена, збільшуються майже експоненційно. Звісно ця проблема не залишається невирішеною, та з'являється все більше інструментів для її подолання.

Однією зі сфер що потребується в таких інструментах найбільше є сфера бізнесу, і в даному випадку мова йде про виробничі та торгівельні підприємства. Данні на таких об'єктах постійно оновлюються, і зі збільшенням обсягу підприємства буде збільшуватися швидкість виникнення ситуацій, які потребують прийняття оперативних, або не дуже, рішень. Одним з основних показників роботи підприємства, на якому базуватиметеся багато рішень, обсяг продажів підприємства.

Кожне підприємство бажає володіти точним прогнозом обсягів своїх продажів. Багато з них покладаються на прогнозування доходів на наступний рік, що часто є відправною точкою для складання річних бюджетів організації. Завдяки розподіленим відділам продажів, компанії тепер більше, ніж будь-коли, покладаються на свою здатність прогнозувати, щоб продовжувати своє зростання та розвиток. Прогнозування продажів – це одночасно наука і мистецтво. Особи, які приймають рішення, покладаються на ці прогнози для планування розширення бізнесу і визначення того, як стимулювати зростання компанії.

Об'єктом дослідження є статистичні дані про обсяг продажів мережі магазинів.

Предметом дослідження є система підтримки прийняття рішень для прогнозування рівнів продажу.

Мета роботи – розробка СППР для аналізу та прогнозування рівнів продажу на підприємствів.

Методом дослідження є побудова прогнозних моделей.

## 1 ОСОБЛИВОСТІ ПРОГНОЗУВАННЯ ПРОДАЖІВ ПІДПРИЄМСТВА

### 1.1 Вплив прогнозу продажів, та фактори що на нього впливають

Прогнозування продажів – це процес оцінки кількості продажів підприємства у майбутній період часу, наприклад місяць, квартал, півріччя, рік. Прогнозування може проводитись як для окремого товару, або групи товарів, так і для усіх товарів проданих певним підрозділом підприємства. Прогноз продажів також може бути визначений як оцінка продажів в грошовому вираженні або в фізичних одиницях в рамках певного плану чи програми, а також в рамках передбачуваного набору зовнішніх ситуацій. Така оцінка є корисним інструментом для менеджменту та керівництва компанії для відстеження результатів роботи та під час прийняття рішень щодо управління [1,2].

В сталих компаніях такі прогнози будуються за результатами продажів у минулому, та підлаштовуються під сучасні ринкові або галузеві тенденції. В свою чергу, у новозаснованих підприємствах, через відсутність достатнього набору історичних даних, прогнозування продажів мають покладатися на інформацію іншого характеру, як наприклад ринкові дослідження, або дані конкурентів.

Потребу та важливість прогнозування продажів можна виділити у багатьох бізнес-процесах [3].

Планування: Наявність прогнозу обсягів продажу, можна скласти бізнес-план або квоту, як для підприємства в цілому, так і для окремих відділів. Наявність таких планів, дозволить планувати роботу компанії на меншому, більш локальному, рівні.

Виявлення потенційних проблем: прогноз рівня продажів дозволяє завчасно виявити та ліквідувати, або пом'якшити наслідки, певних проблем. Наприклад, за допомогою прогнозу, можна виявити що підприємство не вкладається в квартальний план продажів. Детальніше розглянувши проблему

можна буде виявити нестачу покупців, та запустити маркетингову компанію для залучення потенційних клієнтів.

**Управління ресурсами:** Прогнози продажів можуть допомогти підприємству у прийнятті рішень під час найму нових працівників та під час прийняття рішень при управлінні ресурсами підприємства. У разі прогнозу, що вказує на збільшення попиту та обсягу продажів, компанії необхідно буде розподілити свій бюджет для залучення більшої кількості ресурсів, трудових і матеріальних. Якщо ж прогноз вказує на спад попиту, залучення додаткових ресурсів необхідно призупинити або зменшити.

**Управління запасами:** Точніший прогноз продажів дозволяє компанії ефективніше управляти своїми запасами, уникаючи ситуації з нестачею або надлишком. Для виробничих підприємств це напряму сприяє більш стабільній роботі. У разі точного прогнозування попиту, підприємство може краще контролювати свій ланцюг поставок, що, в свою чергу, дозволить в повній мірі користуватися перевагами оперативних замовлень. Стабільність запасів буде також означати стабільність цін, та можливість уникнути розпродажів направлених на позбавлення від надлишку товарів.

**Фінансове планування:** Якісний прогноз продажів надає менеджменту та керівництву компанії інформацію, необхідну для прогнозування прибутку та чистого доходу. Це, в свою чергу, надає гарну можливість для вивчення шляхів їх збільшення. Оскільки розширення і вдосконалення є постійним процесом кожного підприємства, точний контроль прибутку надає кращу можливість контролювання цих процесів.

**Маркетинг:** Прогнозування продажів дає маркетинговому відділу підприємства можливість ефективніше планувати промо-акції, або ж змінити стратегію загалом, якщо виявиться, що продажі будуть слабкими.

На складання прогнозу впливають багато факторів, як внутрішніх для підприємства, так і зовнішніх. До зовнішніх факторів, на які компанія не може впливати на пряму, можна віднести:

Економічні умови: Очевидно, що стан економіки впливає на кожне підприємство та ринок в цілому. Якщо ж настає економічний спад, як підприємства так і покупці стають менш охочими до витрачання грошей, тому і рівні продажів почнуть спадати, що призведе до необхідності до корегування прогнозуючої методики або моделі.

Політичні фактори: Загальнополітичні зміни або ж введення нових законів або положень можуть як допомогти, так і встати на заваді розвитку або функціонування підприємства. Тому, при прогнозуванні продажів на наступний період конче необхідно враховувати і даний аспект.

Галузеві зміни: У кожній галузі промисловості існує аналогічна продукція, вироблена різними компаніями. Такі фактори, як нові технології, дизайн, конкуренти, які проводять рекламні кампанії, або нові компанії, що виходять на ринок, можуть змінитися і вплинути на долю певного підприємства на ринку, що, в свою чергу, вплине на прогнози продажів.

Зміни у продукції: Зміни в продукції підприємства можуть сильно вплинути на прогноз продажів. Ці зміни можуть бути викликані через зміну матеріалів продукції, її дизайну, моделі та ін.

## 1.2 Методи прогнозування продажів

Для вирішення все більшого різноманіття управлінських проблем прогнозування в останні роки було розроблено безліч методів та технік прогнозування продажів, кожен яких застосовується для досягнення окремих цілей, і вимагає розуміння для досягнення кращих результатів. Персонал підприємства, вчасності управлінці та аналітики, також відіграє важливу роль у даному процесі, адже чим краще вони розуміють діапазон можливостей прогнозування, тим більша ймовірність того, що зусилля компанії в галузі прогнозування приведуть до найкращих результатів.

Вибір методу залежить від багатьох факторів - контексту прогнозу, актуальності та доступності історичних даних, бажаної міри точності, періоду часу, протягом якого прогнозується рівень продажів для компанії, а також часу, який є в розпорядженні для проведення аналізу.

Ці фактори повинні зважуватися постійно і на різних рівнях. В цілому, наприклад, аналітики повинні обрати метод, який найкращим чином використовує наявні дані. Якщо аналітик може легко застосувати один метод з задовільною точністю, він не повинен намагатися використати підхід, який потенційно може запропонувати більш високу точність прогнозу, але при цьому вимагає додаткових зусиль для отримання інформації.

У випадку якщо компанія хоче отримати прогноз відносно певного продукту компанії, варто в першу чергу врахувати період існування продукту на ринку, та його поточний етап життя. Від тривалості перебування продукту в обігу напряду впливає обсяг наявної інформації відносно обсягів продажу, що дозволить встановити взаємозв'язки між продажами та зовнішніми факторами.

Прогнозування, в цілому, засноване на загальній передумові збереження даних і подальшому використанні цих даних в якості вхідних для прогнозування обсягів продажу в майбутньому. Якість прогнозу не може бути краще якості даних, на яких він заснований.

Методи, що використовуються для прогнозування продажів, можна розділити по двом категоріям залежно від типу вхідних даних на яких персонал підприємства будуватиме прогноз. Ці категорії отримали назву якісних і кількісних методів прогнозування. Іноді їх ще називають суб'єктивними та об'єктивними методами [4,5].

До якісних методів прогнозування відносяться наступні методи:

- Метод експертних оцінок.
- Метод Делфі.
- Композитний метод.
- Опитування покупців.

- Метод історичної аналогії.
- Думки журі.
- Метод провідних індикаторів.

До кількісних методів прогнозування відносяться такі методи:

- Маркетингові випробування.
- Аналіз часових рядів.
- Метод ковзних середніх.
- Метод експоненціального згладжування.
- Регресійний аналіз.
- Методи інтелектуального аналізу даних.
- Економетричні моделі.

#### 1.2.1 Якісні методи прогнозування продажів

Якісні методи також називають інтуїтивним або суб'єктивними. Ці методи використовуються при потребі побудови довгострокових прогнозів при обмеженому обсязі історичних даних. Додатковими факторами для використання таких методів є присутність можливості змін в урядовій політиці, поява інноваційних рішень у суміжних технологіях виробництва, зміни в структурі споживання та інші.

Якісні методи прогнозування засновані виключно на судженнях експертів у галузі або потенційних клієнтів. Ця категорія в основному включає в себе метод експертних оцінок, метод Дельфі, композитний метод, опитування покупців і метод історичної аналогії.

Метод експертних оцінок є найбільш простим у використанні. Він зазвичай і використовується комерційними організаціями для прогнозування майбутнього попиту на їх продукцію. У цьому методі зазвичай здійснюється

запит до експертів в конкретній області, таких як професіонали в галузі маркетингу, важливі члени ланцюга поставок, а також професійні органи, такі як галузеві асоціації та консультаційні агентства.

Така оцінка не завжди може бути індикатором групової думки. Вона може сприяти більш вокальним учасникам обговорення. Статистична обґрунтованість цього методу також сумнівна.

Таке прогнозування може здійснюватися однією людиною (зазвичай у невеликих компаніях), або ж групою осіб, так званим «Жюрі». У груповому підході, в свою чергу, використовуються два методи: керівники вищої ланки представляють незалежні оцінки без їх обговорення, і вони усереднюються в один прогноз, або ж група проводить засідання, кожна особа представляє окремі оцінки, розбіжності вирішуються, і досягається консенсус.

Метод Делфі є вдосконаленням над методом експертних оцінок. Головна задача методу – визначення подій що можуть настати у майбутньому та ймовірність їх настання.

При використанні цього методу відбирається група експертів та координатор. Експерти в індивідуальному порядку висловлюють координатору свій прогноз. Координатор оцінює надані йому прогнози, поєднує їх та, за необхідності, направляє їх на доопрацювання. Цей процес зазвичай ітерується декілька разів.

Ітерація припиняється при досягненні консенсусу між експертами, або ж при наданні пояснень до розбіжності в їх оцінках. Після цього, координатор проводить статистичний аналіз відповідей, виводячи середні відповіді, варіативність, інтервали прогнозуванн. Тільки координатор матиме доступ до відповідей. Цей процес спрямований на поступове скорочення розбіжностей прогнозів. Прогнози за методом Делфі будуть в основному усередненими. Цей метод є досить оригінальним, і використовується у компаніях, незалежно від їх розміру та спеціальності.

При композитному методі розрахунку торгового зусилля організація опитує свій торговий персонал у роздрібних відділеннях щодо їх прогнозів на

майбутнє. Вважається, що особи, які перебувають в безпосередньому контакті з покупцями, будуть краще проінформовані у питаннях тенденцій попиту на продукт. Після опитування, індивідуальні прогнози об'єднуються для отримання загального прогнозу попиту на певний продукт або групу продуктів для організації. Втім на результати можуть вплинути упередженість персоналу, відсутність інтересу до процесу і незнання більш широких економічних процесів і тенденцій.

Іншим якісним методом прогнозування є опитування клієнтів. Серед регулярних, нерегулярних та потенційних покупців проводиться опитування відносно їх намірів, або ідей, щодо придбання певного товару. В основному, для такого опитування відбирається контрольна група осіб, а інформація отримана від цієї групи оброблюється після проведення опитування. Після цього, отримані дані екстраполюються, з метою отримання загального прогнозу попиту.

Але між заявленими намірами і фактичними покупками завжди будуть існувати розбіжності, розмір яких впливатиме з відповідності контрольної групи до фактичної аудиторії продукту.

Багато компаній часто опитують своїх фактичних або потенційних клієнтів та посередників задля прогнозування ринкового попиту. Деякі компанії використовують споживчі панелі, на яких покупцям надають пробники продукції, або інформацію про неї, і просять надати інформацію про її якість, характеристики, ціну, а також про те, чи будуть вони її купувати. Втім, у багатьох випадках споживачам важко передбачити свої майбутні купівельні звички.

Часто позитивна відповідь, отримана в ході опитування, може не відповідати реальній покупці продукту. Прогнози, засновані виключно на цьому методі, мають тенденцію бути занадто оптимістичними. Втім, цей метод є досить ефективним при малому розмірі цільової аудиторії продукту, або ж якщо покупці беруть на себе серйозні зобов'язання щодо надання своєї думки, наприклад через підписання контракту.



Метод історичної аналогії використовується для прогнозування продажів на нові продукти компанії, історичні данні по яким відсутні у значній кількості, або взагалі відсутні. В такому випадку, для прогнозування використовуватимуть історичні данні продуктів, що мають схожі якості, характеристики, та відносяться до спільної категорії, або лінійки товарів. Таким чином, маркетинговий персонал може використовувати історичну аналогію між двома продуктами і вивести прогноз попиту на новий продукт.

### 1.2.2 Кількісні методи прогнозування продажів

Кількісні, або математичні, методи прогнозування використовуються на основі існуючих даних, історичних, або отриманих навмисно задля побудови прогнозу. Більш складні методи, такі як прогнозування часових рядів і регресійний аналіз, набувають нових варіацій кожен день у міру того, як синоптики вчать їх використовувати за допомогою засобів обробки даних.

Маркетингові випробування є одним з популярних методів вимірювання споживчого сприйняття нових продуктів. Результати тестового маркетингу екстраполюються для складання прогнозів щодо майбутніх продажів. Компанії вибирають обмежене число локацій, які зможуть привести найкращий приклад, для репрезентації цільових споживачів з точки зору демографічних чинників, які включають вік, дохід, стиль життя та покупничі звички.

Продукт стає доступним в торгових точках, а його характеристики виділяються або за допомогою промо-акції, або за допомогою локальної рекламної кампанії. Після цього результати продажів продукту відстежуються за допомогою споживчих досліджень, а зміни, якщо такі є, вносяться перед тим, як перевести продукт до більш глобального ринку.

За іншого підходу компанія відбирають два ринки. Один ринок називається "тестовим", де продукт продається без будь-якої рекламної кампанії, інший, аналогічний, ринок називається "контрольним", і на ньому продукт запускається з проведенням промо-кампанії. Різниця в обсягах продажів на обох ринках є мірою вимірювання ефективності рекламної кампанії. Будь-яка невідповідність з різницею в обсягах продажів на обох ринках є показником розриву між сприйняттям продукту покупцем і характеристиками продукту. Компанії часто змінюють характеристики продукту і тематику його рекламної кампанії в результаті отримання даних про ситуацію на тестовому ринку.

Аналіз часових рядів є серією методик, за допомогою яких будуються прогнозуючі моделі на базі історичних даних компанії, тому збереження такої інформації є критичним для застосування даного методу. Ці дані збираються та інтерпретуються у вигляді співвідношення певного часового періоду (дня, місяця, року) до значення прогнозованої змінної, та значення зовнішніх чинників, які є наявними. Така методика є дуже ефективною при умовах незмінних зовнішніх чинників. Фактори політики підприємства та інформація про конкурентів, зазвичай, не беруться до уваги, якщо їх не можна легко перевести у числовий еквівалент, або якщо вони не несуть суттєвого впливу на прогнозоване значення.

Методи часових рядів використовують хронологічно впорядковані необроблені дані за минулі періоди для прогнозування майбутніх даних. Наприклад, минулі продажі використовуються для прогнозування майбутніх продажів. Однак, через ігнорування зовнішніх чинників, та направленість на використання лише статистичних даних, точність прогнозів зменшується, адже політичні, соціальні та ринкові зміни можуть завдати великого впливу на роботу підприємства.

Незважаючи на те, що цей метод має певні обмеження метод є досить об'єктивним, а дані продажу минулих часів є корисним інформаційним внеском в процедуру прогнозування. Вивчаючи історичну кореляцію рівнів

продажів в часі, менеджер з продажу може виявити тенденції, що дозволять точніше будувати прогнози за допомогою інших методів.

Будуючи прогнози за допомогою часових рядів, варто звернути увагу на деякі суттєві зміни. Такі зміни можна розділити на чотири категрії.

Зміни, які відбулися в результаті загальної тенденції даних до зростання або спадання, називаються світськими рухами. Зміни, які відбулися протягом 12 місяців в результаті зміни клімату, погодних умов або фестивалів, називаються сезонними коливаннями. Зміни, які відбулися в результаті загальноекономічного росту або спаду, називаються циклічними змінами. Зміни, що відбулися в результаті таких непередбачуваних сил, як повені, землетруси, голод та інше, класифікуються як нерегулярні або непередбачувані зміни.

У разі зміни дії компанії у вигляді рекламних кампаній, надлишку стимулюючих продажів, зміни цін, відкриття нових точок продажів або виявлення нового використання продукту, будь-яка така зміна може вплинути на часовий ряд і тенденція може стрімко змінитися, тим самим знижуючи рівень точності методу, використововуваного для прогнозування.

Моделі всіх часових рядів шукають закономірності в даних минулих періодів. Чотирма компонентами даних минулих періодів, що мають відношення до аналізу часових рядів, є тренди, сезонність, циклічність і нерегулярність.

Тренди є основою руху в часовому ряді. Наприклад, в разі зростання чисельності населення на 1 відсоток, ймовірно, буде спостерігатися тенденція до збільшення попиту на основні ходові товари. Тренд може рухатися вгору або вниз в залежності від зусиль по розробці продукту, споживчих смаків, змін в технологіях, широких економічних тенденціях і інших фундаментальних змін на ринку цього продукту.

Циклічні зміни – це зміни, які відбувається протягом періоду, що перевищує один рік. Певні зміни в економіці, такі як зміна банківських процентних ставок, як правило, відбуваються протягом періоду, що перевищує

один рік. Житлова, автомобільна та туристична галузі є прикладами впливу циклічності. Покупці, швидше за все, будуть купувати ці товари або послуги в залежності від коливань банківських процентних ставок.

Нерегулярність – це випадкові зміни у структурі продажів. Природні катаклізми, різкі соціальні або політичні зміни відбуваються випадковим чином і впливають на структуру попиту. Нерегулярні події можуть здійснити істотний вплив на структуру продажів.

Найбільш популярними методами прогнозування часових рядів є прогнози трендів, метод ковзних середніх, метод експоненціального згладжування, метод регресії.

Метод інтелектуального аналізу даних дуже схожий до методу аналізу часових рядів. В певній мірі, він є покращенням цього методу, адже їх головний принцип однаковий – використання історичних даних, виявлення у них закономірностей, та їх подальше використання для побудови моделей процесу та прогнозу.

Головна відмінність між цими методами це підхід до побудови моделей процесів. При аналізі часових рядів, для побудови моделей використовуються суто математичні методи, а той час як при інтелектуальному аналізі даних моделі будуються за допомогою штучних нейронних мереж.

## Висновки до розділу

У розділі було розглянуто важливість прогнозування рівнів продажу для підприємства. Було розглянуто інформацію та вплив, який прогноз може надати, а також фактори що на нього впливають. Також проведено огляд існуючих та використовуваних, методів прогнозування продажів. Було

детально розглянуто їх недоліки та переваги, а також ситуації, в яких варто використовувати той чи інший прогноз.

## 2 ВИБІР МАТЕМАТИЧНИХ МЕТОДІВ ДЛЯ ПОБУДОВИ МОДЕЛЕЙ ТА ПРОГНОЗУВАННЯ ПРОДАЖІВ

### 2.1 Моделі що використовуються для опису та прогнозування часових рядів

Як було зазначено раніше, регресійний аналіз та аналіз часових рядів є одним із найпопулярніших, та досить ефективних підходів до прогнозування. Розглянемо декілька варіацій регресійних моделей, які можуть використовуватися для прогнозування продажів на підприємстві [6].

#### 2.1.1 Модель авторегресії

Авторегресійна (АР) модель – модель часових рядів, за якою кожне значення ряду залежить від попередніх значень цього ж ряду лінійно. Загальний вигляд авторегресійного процесу порядку  $p$ :

$$X_t = c + \sum_{i=1}^p a_i * X_{t-i} + \varepsilon_t,$$

де  $a_i$  – коефіцієнти авторегресії;

$c$  – константа;

$\varepsilon_t$  – білий шум або похибка моделі.

Модель АР(3) матиме наступний вигляд:

$$y(k) = a_0 + a_1 y(k-1) + a_2 y(k-2) + a_3 y(k-3).$$

Порядок використання автрегресійної складової можна визначити за допомогою ступені автокореляції змінної. Значення автокореляційної функції  $r_y(s)$  обчислюється за наступною формулою:

$$r_y(i) = r_{y(k)y(k-i)} = \frac{1}{n-1} \frac{\sum_{k=i+1}^n \{[y(k) - \bar{y}][y(k-i) - \bar{y}]\}}{\sigma_y^2},$$

де  $n$  – розмір вибірки;

$\sigma_y^2$  – дисперсія змінних  $y(k)$  на вибірці;

$i = 1, 2, 3, \dots$

Отримавши значення АКФ, залежно від вимог до моделі, обирається граничне значення функції, та відбираються лаги, для яких це значення було перевищене. Кожен такий лаг відповідатиме значущому порядку АР складової в моделі.

Точніше порядок АР складової можна визначити за допомогою часткової автокореляційної функції (ЧАКФ). ЧАКФ визначає прямий вплив одного значення ряду на інший, відкидаючи кореляційних вплив проміжних значень. Аналогічно до АКФ, розмірність АР складової моделі може визначатися відповідно до ненульового коефіцієнта ЧАКФ у найбільшому найбільшим лагом.

### 2.1.2 Модель ковзного середнього

Ковзне середнє (КС) представляє собою набір значень, що відображають середнє значення ряду на певному проміжку, і використовується для відображення тенденцій процесу, та згладження його несуттєвих коливань.

Загальна формула для ковзного середнього:

$$MA(k) = \frac{\sum_{i=1}^N y(k-i+1)}{N},$$

де  $N$  – розмір вікна КС ;

$y$  – часовий ряд вхідних даних

Власне КС компонента моделі матиме вигляд:

$$y(k) = mv(k) + \sum_{j=1}^q b_j * mv(k-j),$$

де  $mv(k)$  – значення ковзного середнього для  $k$ -го елемента часового ряду;

$b_j$  – коефіцієнт моделі.

### 2.1.3 Модель авторегресії з ковзним середнім (АРКС)

Авторегресійна модель з ковзним середнім (АРКС) використовується для аналізу та прогнозування стаціонарних часових рядів. Фактично, така модель скрадатиметься з двох попередньо зазначених моделей АР та КС.

Моделлю АРКС( $p,q$ ), де  $p$  та  $q$  порядки авторегресійної складової та ковзного середнього відповідно, називається наступна модель:

$$X_t = c + \sum_{i=1}^p a_i * X_{t-i} + mv_t + \sum_{i=1}^q b_i * mv_{t-i} + \varepsilon_t,$$

Оскільки АРКС модель являє собою комбінацію АР та КС моделей, її структура ускладнюється, а отже і виростає і кількість необхідних розрахунків, і збільшується потреба у розрахункових потужностях обчислювальної



системи. Втім, через складнішу структуру, також і зменшується потреба у більшій кількості використовуваних моделлю коефіцієнтів.

## 2.2 Методика побудови моделей нестационарних процесів

Одна з основних методик побудови моделей часових рядів Була запропонована Д. Боксом та Г. Дженкінсом. Вона покладається на використання АРКС та АРІКС моделей, для прогнозування фінансових та економічних процесів [7].

В рамках даної роботи, цю методику можна інтерпретувати наступним чином:

- Виконати аналіз процесу, для якого будується модель.
- Провести попередню обробку даних.
- Виконати аналіз наявності нелінійності в процесі.
- Побудувати та відібрати моделі-кандидати, та обрати їх структуру.

### 2.2.1 Аналіз процесу та попередня обробка даних

Важливість даного етапу полягає у тому, що якісний аналіз процесу дозволяє побудувати більш якісні моделі, а попередня обробка даних необхідна для цього в цілому. Основними діями під час аналізу даних, є встановлення розмірності процесу, та визначення зв'язків між використовуваними даними. У випадку з моделями авторегресії, це означатиме розрахунок значень АКФ та ЧАКФ. Також варто розглянути можливість розкладу процесу на простіші підпроцеси, які можна описувати окремо.

Етап аналізу також включає огляд літератури, що описує даний процес, а також огляд уже розроблених моделей процесу, та пошуку можливості їх інтеграції у власну модель. Це може значно спростити вибір структури власної моделі, і підвищити її точність.

Попередня обробка даних необхідна для полегшення подальшої роботи з ними, як під час аналізу даних, так і при оцінці параметрів моделі. Серед методів попередньої обробки даних видаляють нормування, корегування та інші способи отримання нових даних, на базі існуючих.

### 2.2.2 Відбір структур моделі

Поняття структури моделі включає в себе її компоненти, порядок цих компонентів, вимірність, можливі нелінійності та їх тип.

Приклад вибору порядку компонентів моделі наведено в пунктах 2.1.1 та 2.1.2 для компонент AP та KC відповідно.

Якщо ж обрана модель не є авторегресійною, або вона міститиме екзогенні, тобто зовнішні, змінні, необхідно буде визначити ступінь кореляції між змінними, та прийняти рішення відносно їх включення до моделі, для цього використовується кореляційна функція та матриця.

Кореляційна матриця дозволяє визначити існування зв'язку між декількома змінними. Розглянемо кореляційну матрицю для випадку трьох змінних  $x$ ,  $y$ ,  $z$ :

$$R = \begin{bmatrix} r_{yy} & r_{xy} & r_{zy} \\ r_{yx} & r_{xx} & r_{zx} \\ r_{yz} & r_{xz} & r_{zz} \end{bmatrix},$$

де  $r_{xy}$  – коефіцієнт кореляції між змінними  $x$  та  $y$ ,  $r_{xy} = r_{yx}$  і т.д.

Для обчислення елементів кореляційної матриці необхідно мати часові ряди усіх змінних, які синхронізовано в часі. Формула для знаходження коефіцієнтів кореляції виглядає наступним чином:

$$r_{yx} = \frac{\frac{1}{N-1} \sum_{k=1}^N \{[x(k) - \bar{x}][y(k) - \bar{y}]\}}{\sigma_x \sigma_y},$$

де  $\bar{x}, \bar{y}$  – середні значення відповідних змінних на обраному часовому ряді;

$N$  – розмір часового ряду;

$K$  – дискретний час спостереження значення змінної;

$\sigma_x, \sigma_y$  – стандартні відхилення відповідних змінних (квадратні корені дисперсії)

$$\sigma_y = \sqrt{\sigma_y^2} = \left[ \frac{1}{N-1} \sum_{k=1}^N [y(k) - \bar{y}]^2 \right]^{1/2},$$

На основі знайдених значень кореляції приймається рішення про включення, або не включення відповідних змінних у модель.

Якщо ж у процесі наявні незначні коливання, які необхідно було б згладити, або процес має незначні тенденції до зміни, варто розглянути можливість включення в модель компонент ковзної середньої. Саме ковзне середнє і відповідна компонента моделі були розглянуті раніше. Варто також зазначити, що ковзне середнє, залежно від потреб процесу та моделі, можуть будуватися на значеннях самої оцінюваної змінної, або ж на базі залишків моделі.

### 2.2.3 Оцінювання параметрів моделей

Після відбору моделей кандидатів та їх структури необхідно оцінити параметри цих моделей. Зробити це можна за допомогою багатьох методів, одним з яких є метод найменших квадратів (МНК) та його рекурсивний аналог (РМНК).

Обрані моделі фактично являють собою систему  $n$  рівнянь з  $m$  змінними, де  $n$  – розмір вибірки, а  $m$  – кількість параметрів усіх компонентів моделі, та  $m < n$ .

Система рівнянь у матричному вигляді може бути записана як  $y = X\theta$ , де  $y$  – вектор значень часового ряду шуканої змінної;  $X$  – матриця екзогенних змінних;  $\theta$  – вектор параметрів моделі.

Так наприклад для моделі  $AP(1)$  матриця вимірів та вектор параметрів моделі виглядатимуть наступним чином:

$$X = \begin{bmatrix} 1 & y(2) \\ 1 & y(3) \\ \vdots & \vdots \\ 1 & y(n-1) \end{bmatrix}; \quad \theta = \begin{bmatrix} a_0 \\ a_1 \end{bmatrix},$$

Отже для знаходження параметрів залишається розв'язати систему у матричному вигляді відносно  $\theta$ . Оцінки МНК знаходяться за допомогою наступного виразу:

$$\hat{\theta} = [X^T X]^{-1} X^T y.$$

### 2.3 Критерії адекватності моделей

Після побудови моделей-кандидатів, необхідно відібрати з них найкращі, а отже, необхідно скористатися критеріями, що допомагають оцінити ті, або інші показники моделі.

Не існує якихось стандартних методиг для оцінки якості моделі, окремі критерії та показники використовуються за потреби. Наприклад, перевірити адекватність моделі можна проаналізувавши залишки від прогнозу, побудованого моделлю.

Також допускається використання візуального аналізу за допомогою графіків прогнозів та похибки моделі. В такому випадку, варто звернути увагу на екстремальні значення, та монотонні проміжки.

Іншим показником адекватності моделі, який отримується за допомогою похибок, кореляція між значеннями похибок моделі, в цьому випадку відсутність кореляції буде казати про вищий ступінь адекватності моделі. Для аналізу кореляцій можна використовувати АКФ та ЧАКФ.

Критерій Дарбіна-Уотсона використовується для перевірки автокореляції першого порядку між елементами часового ряду похибок моделі. Для встановлення корельованості похибок достатньо скористатись формулою:

$$DW = \frac{\sum_{t=2}^N (\varepsilon_t - \varepsilon_{t-1})^2}{\sum_{t=1}^N \varepsilon_t^2},$$

де  $\varepsilon_t$  – значення похибки на  $t$ -му періоді;

$N$  – кількість періодів.

Граничні значення критерію становлять 0 та 4, при кореляціях 1 т -1 відповідно. Ідеальним значенням вважається  $DW = 2$ , так як кореляція буде

відсутня повністю, проте на практиці наближення значення критерію до 2 вважається достатнім показником відсутності кореляції похибок.

Головним недоліком даного критерію є його неточність на малих вибірках даних. Іншим недоліком цього критерію є його неефективність при використанні до моделей авторегресії. Для перевірки кореляції залишків у АР процесах варто скористатися h-критерієм, або h-статистика Дарбіна.

Коефіцієнт множинної детермінації  $R^2$ , який обчислюється так:

$$R^2 = \frac{\text{var}(\hat{y})}{\text{var}(y)} = 1 - \frac{SSE}{SST},$$

де  $\text{var}(\hat{y})$  – дисперсія залежної змінної, оціненої за допомогою побудованої моделі;

$\text{var}(y)$  -дисперсія вимірів залежної змінної;

$$SSE = \sum_{k=1}^N [y(k) - \hat{y}(k)]^2 - \text{сума квадратів похибок моделі};$$

$$SST = \sum_{k=1}^N [y(k) - \bar{y}]^2 - \text{загальна сума квадратів};$$

$\bar{y}$  – середнє значення.

Найкращим значенням є  $R^2 = 1$ , тобто повна збіжність між оціненим значенням цільової змінної, та її реальним значенням. Цей параметр можна трактувати, також, як міру інформативності моделі, якщо вибрати за міру інформативності дисперсію. Таким чином,  $R^2$  показує рівень інформативності моделі по відношенню до інформативності вибірки даних, за допомогою якої вона була оцінена.

Іншим показником, що можна використати для оцінки моделей є звичайна сума квадратів похибок. Сума квадратів похибок для вибраної моделі повинна бути мінімальною, тобто,

$$\sum_{k=1}^N e^2(k) = \sum_{k=1}^N [\hat{y}(k) - y(k)]^2 \rightarrow \min,$$

у порівнянні з усіма іншими моделями. Аналогічно, можна використати суму коренів квадратів похибки, та корінь усередненого квадрата похибки.

Також для оцінки моделі використовують інформаційний критерій Акайке

$$AIC = N \ln \left( \sum_{k=1}^N e^2(k) \right) + 2n,$$

де  $n = p + q + 1$  – число параметрів моделі ( $p$  – число параметрів авторегресійної частини моделі;  $q$  – число параметрів ковзного середнього);

$N$  – розмір вибірки даних.

Даний критерій містить у собі суму квадратів похибок моделі, це означає що найкращим значенням критерію є найменше значення. Втім, додатковою перевагою даного критерія є його орієнтованість на структуру самої моделі, та кількість параметрів що у ній використовуються. Введення нового регресора приводить до збільшення критерію (при цьому збільшується  $n$ ), але, разом з тим, зменшується сума квадратів похибок і критерій в цілому зменшується. Якщо регресор не покращує модель, то критерій збільшується.

## 2.4 Нейронні мережі що використовуються для прогнозування часових рядів

В останні роки нейронні мережі стали однією з найбільш популярних тенденцій в машинному навчанні і знайшли своє застосування в багатьох областях, включаючи і фінансове прогнозування.

Штучна нейронна мережа – це математична модель, а також її програмна реалізація. Штучні нейронні мережі є одним з основних інструментів, що використовуються в машинному навчанні. Як впливає з "нейронної" частини їх назви, це системи, натхненні людським мозком, які призначені для відтворення процесу навчання людини.

Нейронні мережі складаються з вхідного і вихідного шарів, а також, в більшості випадків, прихованого шару. Шари нейромережі складаються з нейронів, які являють собою функції які перетворюють вхідні дані у вихідні. Увесь процес навчання полягає у корегуванні внутрішніх коефіцієнтів нейронів, а також коефіцієнтів, які пов'язують нейрони між собою, базуючись на тому, наскільки вихідні дані нейромережі задовольняють дійсність. Нейромережі є відмінними інструментами для пошуку патернів, які занадто складні або численні для того, щоб їх можна було запрограмувати вручну.

Нейронна мережа складається з нейронів, з'єднаних один з одним. За кожним таким зв'язком закріплено ваговий коефіцієнт, який вказує на важливість цього взаємозв'язку між нейронами.

Кожен нейрон має функцію активації, яка визначає вихід нейрона. Функція активації використовується для введення нелінійності у можливості моделювання мережі. Існує декілька видів функції активації, які використовуються залежно від структури нейронної мережі та даних з якими вона працює.

Навчання нашої нейронної мережі, тобто оптимізація значень параметрів є найбільш важливою частиною машинного навчання, і цей процес можна розглядати як ітераційний алгоритм прямого та зворотнього поширення похибки шарами мережі.

Перша фаза прямого поширення відбувається, коли у мережу подаються дані для навчання, і вони проходять усю нейронну мережу, генеруючи шукані вихідні дані. Тобто відбувається передача вхідних даних через мережу таким чином, що усі нейрони трансформують інформація, яку вони отримують від



нейронів попереднього шару, і передають її нейронам наступного шару. Коли дані пройдуть усі шари мережі, будуть отримані відповідні вихідні дані.

Після цього, застосовується функція втрат, яка вимірює похибку, та оцінює наскільки ефективним був наш прогноз по відношенню до реального результату. Такий тип навчання називається навчанням з учителем.. В ідеалі ми хочемо, щоб наша похибка була нульовою, тобто без розбіжностей між оціненими та реальними даними. Тому, коли модель тренується, ваги взаємозв'язків нейронів будуть поступово коригуватися, поки не будуть отримані хороші оцінки.

Після обчислення функції втрат, у разі якщо бажане значення похибки не було досягнуто, за допомогою одного з методів оптимізації обчислюється нове значення параметрів нейронів та вагів.

#### 2.4.1 Багатошаровий перцептрон

Багатошаровий перцептрон набув широкої популярності через широкі можливості його застосування. Обчислювальний інтелект, як наука досліджує, як прості моделі біологічних мізків можуть бути використані для вирішення складних обчислювальних задач, наприклад задач прогнозування або моделювання. Кінцевою метою є не створення реалістичних моделей мозку, а натомість розробка надійних алгоритмів та структур даних, які можуть бути використанні для вирішення складних проблем.

Як було зазначено раніше, головна перевага нейронних мереж походить від їх здатності навчатися, розпізнавати особливі ознаки у даних та співвідносити їх із вихідними даними. У цьому сенсі нейронні мережі займаються задачами зіставлення вхідних даних з вихідними. Нейромережі здатні відтворити будь-яку математичну функцію відображення і були підтверджені як універсальне рішення для вирішення задач апроксимації.

Можливість вирішувати складні задачі нейронні мережі беруть на пряму від своєї ієрархічної, або багат шарової, структури. Багат шаровість дозволяє нейромережі визначати ознаки модельованого процесу або функції на різних рівнях, та у подальшому, об'єднуючи ці ознаки, перейти до більш складніших структур. Наприклад від рівнянь або ліній, до набору рівнянь або фігур.

Складовим елементом нейронних мереж є штучні нейрони. Це прості обчислювальні одиниці, які приймають зважені вхідні сигнали, оброблюють їх, та видають на наступний шар вихідний сигнал за допомогою функції активації.

Принцип роботи нейрону є схожим з рівнянням регресії, де на кожну змінну (вхід нейрону) припадає коефіцієнт (ваговий коефіцієнт). Як і регресія, кожен нейрон також має зміщення, яке можна розглядати як вхід, який завжди має значення 1, і його теж слід зважувати. Наприклад, нейрон може мати два входи, і в цьому випадку йому потрібно три ваги. Один для кожного введення та один для зміщення. Ваги часто ініціалізуються як невеликі випадкові значення, наприклад значення в діапазоні від 0 до 1, хоча можна використовувати і більш складні схеми ініціалізації. [8]

Пов'язані між собою нейрони і є основою кожної нейромережі. Нейрони що поєднані у ряд, та отримують однаковий, або ж аналогічний, набір даних на вході складають шар нейромережі. Архітектуру нейронів у мережі часто називають мережевою топологією.

Перший, або нижній, шар, що приймає дані безпосередньо з набору даних, називається вхідним, або видимим, шаром, оскільки це відкрита частина мережі. В основному, вхідний шар мережі зображується з одним нейроном на кожне вхідне значення або вектор з початкових даних. Нейрони вхідного шару не являють собою повноцінні нейрони, які було описано вище. Максимум що вони можуть робити – це перетворювати вхідні, необроблені данні, у більш корисний вигляд, який буде використовуватися наступними шарами, наприклад нормалізувати данні.

Шари що йдуть безпосередньо після вхідного шару, називаються прихованими шарами, оскільки з ними відсутня можливість прямої взаємодії. Найпростіша мережева структура являє собою єдиний нейрон у прихованому шарі, який безпосередньо виводить значення. Враховуючи приріст обчислювальної потужності та появу ефективних бібліотек та платформ машинного навчання, можна створити дуже глибокі нейронні мережі. Під нейронною мережею глибокого навчання мають на увазі мережі, які містять у собі декілька прихованих шарів. Такі мережі називають глибокими тому, що без технологічних досягнень, наявних сьогодні, навчання такої мережі зайняло б дуже велику кількість часу.

Останнім шаром нейронної мережі є вихідний шар, і відповідає він безпосередньо за виведення значення, або набору значень, у бажаному в умовах задачі форматі. Функція активації вихідного шару сильно обмежена типом поставленої задачі. Так можуть використовуватися для задач регресії вихідний шар буде складатися з одного нейрону без функції активації.

Після того як структура нейронної мережі була обрана, починається процес навчання. Першим кроком буде відбір та підготовка набору даних. Обов'язковим є перетворення даних у числовий формат, так як нейронні мережі не працюють з текстовими або медіа файлами напряму. Наприклад у випадку задач класифікації зображень, вхідні данні конвертуються у чисельний формат. Одним із способів такої конвертації є розбиття зображення на сегменти(наприклад пікселі), де кожен сегмент буде представлений 4ма числами. Ці числа будуть відповідати за домінуючий колір сегменту, та вказувати на рівень червоного, синього та зеленого кольорів та яскравість. Таким чином, зображення можна представити у вигляді матриці розміром  $4 \times n$ , де  $n$  – кількість виділених сегментів у зображенні.

Класичний і досі переважний алгоритм тренування для нейронних мереж називається стохастичним градієнтом. При такому алгоритмі один набір даних за раз подається на мережу у якості входу. Мережа покроково проводить дані через шари нейронів, і, нарешті, генерує вихідне значення. Це

називається передачею даних вперед по мережі. Такий тип передачі даних використовується після завершення процесу її навчання для генерації прогнозів.

Після цього, отриманий вихід мережі порівнюється з очікуваним виходом, та обчислюється значення похибки. Після цього, починається процес оберненого поширення інформації та корегування параметрів мережі.

#### 2.4.2 Метод зворотнього поширення похибки

Backpropagation, або "метод зворотнього поширення похибки" - це алгоритм для навчання штучних нейронних мереж. Залежно від структури нейромережі та функції помилок, метод обчислює градієнт функції помилки відносно ваг нейронної мережі. Це узагальнення дельта-правила для перцептронів до багатошарових нейронних мереж.

Зворотність впливає з того, що обчислення градієнта відбувається в оберненому напрямку поширення інформації через мережу, при цьому градієнт кінцевого шару ваг розраховується першим, а градієнт першого шару ваг підраховується останнім. Розрахунок градієнта з одного шару повторно використовуються при обчисленні градієнта попереднього шару. Цей зворотний потік інформації відносно помилки дозволяє ефективніше знаходити градієнт на протиставлення до наївного методу, за якого градієнти кожного шару знаходяться відокремлено.

Даний метод набув нової популярності, враховуючи широке застосування нейронних мереж для розпізнавання зображень та мовлення. Обернене поширення похибки вважається ефективним алгоритмом, а сучасні реалізації використовують переваги спеціалізованих графічних процесорів для подальшого підвищення продуктивності.

Так як метод оберненого поширення похибки є аналогом дельта-правила для багатошарових мереж, для його реалізації необхідні виконання наступних речей: наявність вихідних даних для вибірки навчання, мережа має розповсюджувати інформацію лінійно та містити параметри що мають оптимізуватися (в основному це вагові коефіцієнти), а також наявність визначеної функції витрат (похибки).

Для оптимізації параметрів нейронної мережі при використанні методу *backpropagation* можуть використовуватися різні методи. Одними з найпопулярніших є метод градієнтного спуску та метод спряжених градієнтів.

Розглянемо детальніше процес навчання нейронної мережі з використанням алгоритму *backpropagation*.

1. Ініціалізація вагових коефіцієнтів та зсуву.

Зазвичай, ваги нейронної мережі ініціалізуються невеликими, випадково згенерованими числами. Ініціалізація ваг нейронних мереж - це ціле поле дослідження, оскільки ретельний підхід до ініціалізації мережі може значно пришвидшити процес навчання мережі. Сучасні бібліотеки глибокого навчання, такі як Keras, пропонують безліч методів ініціалізації мережі.

2. Поширення сигналу від вхідних нейронів до нейронів прихованого шару.
3. Поширення сигналу від нейронів прихованого шару до нейронів вихідного.

Кожен нейрон прихованого (та вихідного) шару просумовує та зважає усі вхідні сигнали, та додає зсув. Після цього, отриманий результат проходить через функцію активації даного нейрону і передається на наступний шар. Цей процес має наступний вигляд:

$$y_{in_{ik}} = \sum (y_{out_{j(k-1)}} * w_{jk}) + b_{ik};$$

$$y_{out_{ik}} = f_k(y_{ik}),$$

де  $y_{in_{ik}}$  – вхідне значення  $i$ -го нейрону шару  $k$ ;  
 $y_{out_{ik}}$  – вихідне значення  $i$ -го нейрону шару  $k$ ;  
 $w_{jk}$  – ваговий коефіцієнт;  
 $b_{ik}$  – значення зсуву;  
 $f_k$  – функція активації нейронів  $k$ -го шару.

#### 4. Розрахунок похибки.

Після отримання сигналу з вихідного шару, розраховуються значення похибки для кожного виходу, а також загальне значення функції втрат. Навчання зупиняється якщо виконується одна з умов зупинки. Якщо жодна з умов зупинки не виконується необхідно оновити значення параметрів мережі, та перейти на нову ітерацію

#### 5. Оновлення параметрів мережі.

При використанні алгоритму градієнтного спуску оновлення параметрів виглядає наступним чином:

$$\theta^{i+1} = \theta^i - \alpha * \partial \theta,$$

де  $\theta^i$  – значення параметру на  $i$ -й ітерації;

$\alpha$  - коефіцієнт навчання мережі;

$\partial \theta = \frac{\partial C}{\partial \theta}$  - градієнт функції вартості відносно параметра  $\theta$ .

Складність оновлення параметрів полягає у знаходженні градієнту для кожного параметру. Проте в основу методу оберненого поширення похибки закладено механізм що значно спрощує цю задачу. Розглянемо цей процес на конкретному прикладі вихідного шару нейронної мережі. Для вихідного сигналу  $y_{out}$  нейронів вихідного шару розраховується значення похибки, які поєднуються у функцію втрат  $C$ . У нейроні сигнали поєднуються та проходять через функцію активації. На входи нейрону подається зважений сигнал зі зміщенням -  $y_{in}$ .

Необхідно знайти  $\frac{\partial C}{\partial w}$ . Враховуючи потенційну складність розрахунків у нейронах знайти часткову похідну напряму може бути неможливо. Проте, ми можемо розділити це підрахунок на три кроки використавши правила диференціювання складної функції.

$$\frac{\partial C}{\partial w} = \frac{\partial C}{\partial y_{out}} \frac{\partial y_{out}}{\partial y_{in}} \frac{\partial y_{in}}{\partial w}.$$

Ця задача вже є значно простішою. Розглянемо кожну похідну окремо.

Так як

$$y_{in}^k = y_{out}^{k-1} * w + b,$$

$$\text{тоді } \frac{\partial y_{in}}{\partial w} = y_{out}^{k-1}.$$

Значення  $\frac{\partial y_{out}}{\partial y_{in}}$  напряму залежить від функції активації нейрону.

Наприклад для випадку використання сигмоїди похідна матиме вигляд

$$\sigma(y_{in})(1 - \sigma(y_{in})),$$

а для функції активації ReLU матиме значення 1 або 0, залежно від самого сигналу.

Очевидно що значення похідної  $\frac{\partial C}{\partial y_{out}}$  також залежить від самі функції

втрат. Наприклад для квадратичної функції втрат:

$$\frac{\partial}{\partial y_{out}} \sum (y_{real} - y_{out})^2 = 2(y_{real} - y_{out}).$$

Поєднавши усі похідні отримуємо:

$$\frac{\partial C}{\partial w^k} = y_{out}^{k-1} * \sigma(y_{in}^k) (1 - \sigma(y_{in}^k)) * 2(y_{real} - y_{out}^k).$$

Маючи усі значення дуже легко оновити параметр  $w^k$ .

Керуючись даною логікою та правилом диференціювання складної функції аналогічно, на будь-якому шарі, знаходяться інші параметри що потребують оновлення.

### 2.4.3 Нейронні мережі РБФ

Один з видів нейромереж є мережа Радіально Базисних функцій. Її загальне представлення було сформульовано у 1988 році Брухмедом і Лоу. Даний тип мереж має багато застосувань, зокрема вони можуть використовуватися для задач апроксимації функції, прогнозування часових рядів, задач класифікації та керування системою [9].

Мережі РБФ краще всього використовувати для задач апроксимації та прогнозування, при яких відсутня необхідність знаходити та визначати «глибокі» параметри модельованих функцій, тобто відсутня потреба у глибинному навчанні. Даний тип нейромереж зарекомендував себе як універсальний апроксиматор. А для задач прогнозування часових рядів може цілком конкурувати з багатошаровими перцептронами що використовують метод зворотнього поширення похибки. При цьому, мережа є значно простішою у реалізації та навчанні, і є менш вимогливою до продуктивності машини на якій вона виконується.

Мережі RBF зазвичай мають лише один прихований шар, та, очевидно, шари входу та виходу, та їх процес навчання вимагає наявності «учителя»,



тобто реальних вихідних даних що відповідають кожному набору вхідних даних. На вхід мережі зазвичай подається вектор значень, що відповідає одному прикладу з вибірки вхідних даних. Свою назву мережа отримала через використання радіально базисних функцій в якості функцій активації прихованого шару. Вихід мережі матиме наступний вигляд:

$$\varphi(x) = \sum_{i=1}^N \omega_i p(\|x - c_i\|),$$

де  $\omega_i$  – ваговий коефіцієнт  $i$ -го нейрону;

$N$  – загальна кількість нейронів прихованого шару;

$c_i$  – центральний вектор нейрону  $i$ -го шару;

$p$  – радіально базисна функція.

Данні функції називаються радіально базисними так як вони є симетричними відносно свого центру, і значення функції залежить лише від відстані до нього. За норму, або відстань, зазвичай приймається евклідова відстань. Зазвичай в якості радіально базисної функції береться модифікована гаусіана:

$$p(\|x - c_i\|) = e^{(-\beta_i \|x - c_i\|^2)}.$$

Параметри  $\beta$  та  $c_i$  можуть бути визначені ще до початку навчання нейромережі, ваговий коефіцієнт же ініціалізується одним з методів, наприклад випадково, та підлягає подальшій оптимізації.

Існує декілька методів ініціалізації параметра  $c_i$ . Одним з найпопулярніших вважається ініціалізація за допомогою кластеризації. В цьому випадку вхідні данні розбиваються на кластери, та центри РБ функцій обираються як центри кластерів. В якості методу кластеризації можна використовувати методи  $k$ -середніх, карти Кохонена та інші.

Втім, розташувати центри можна і випадковим чином, проте це дає менш ефективні результати навчання. Задавши кількість шуканих кластерів, або класів, метод к-середніх не тільки допоможе визначити центри кожного класу, а і надасть інформацію щодо кількості вхідних наборів даних що входять до цього класу. Це також допоможе нам задати значення параметру  $\beta$  що відповідає за зсув радіально-базисної функції. Знайти його значення можна наступним чином:

Знаходимо середню відстань від точок, що належать одному з кластерів:

$$\sigma = \frac{1}{m} \sum_{i=1}^m \|x - c_i\|.$$

де  $m$  – кількість точок потрапивших у кластер.

Значення  $\beta$  отримуємо наступним чином:

$$\beta = \frac{1}{2\sigma^2}.$$

Ініціалізація центрів РБ функцій є першим кроком у навчанні нейромережі. Цей крок відбувається без учителя. Цілком раціональним під час навчання мережі буде автоматична генерація та навчання декількох моделей кандидатів, з різною кількістю центрів РБ функцій, для подальшого відбору найкращої моделі.

Ініціалізувавши значення параметрів та початкове значення вагів, можна перейти до процесу навчання мережі. В якості функції витрат може використовуватися середньоквадратична похибка.

Процес навчання мережі, вчасності оптимізації вагів напряду залежить від її структури. Якщо ж параметри радіально базисних функцій (центр та радіус) зафіксовані, і мережа має лінійну структуру, оптимізувати вагові коефіцієнти можна за допомогою псевдо обернених матриць із рівняння:

$$\begin{bmatrix} g_{11} & \cdots & g_{1m} \\ \vdots & \ddots & \vdots \\ g_{n1} & \cdots & g_{nm} \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_m \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix},$$

де  $g_{ij}$ - значення  $j$ -ї РБФ для  $i$ -го входу. Тоді знаходимо інверсію добутку матриці  $G$  на транспоновану матрицю  $G$ :

$$A^{-1} = (H^T H)^{-1},$$

і далі знаходимо остаточне значення вектору вагів  $W$  за формулою:

$$W = A^{-1} H^T Y.$$

Цей метод є можливим, так як для лінійної РБФ мережі, на відміну від багатошарових перцептронів, існує унікальний локальний мінімум.

Якщо ж параметри РБФ не фіксовані, тобто крім вагів необхідно налаштувати параметри функції активації кожного нейрона (зміщення функції і її радіус), мережа стає нелінійною. І вирішувати задачу оптимізації доводиться з використанням ітеративних чисельних методів оптимізації, наприклад за градієнтних методів. В такому випадку метод навчання є наближеним на Backpropagation, за тим виключенням що застосовується він лише до одного шару.

Останнім часом набули популярності методи навчання RBF мережі, в яких використовується поєднання генетичних алгоритмів для підбору параметрів активаційних функцій і методів лінійної алгебри для розрахунку вагових коефіцієнтів вихідного шару. Тобто на кожній ітерації пошуку генетичний алгоритм самостійно вибирає в яких точках простору вхідних сигналів мережі розмістити центри РБ функцій нейронів прихованого шару і призначає для кожної з них ширину вікна. Для отриманої таким чином

сукупності параметрів прихованого шару обчислюються ваги вихідного шару і отримується помилка апроксимації, яка служить для генетичного алгоритму індикатором того, наскільки підходящим є даний варіант. На наступній ітерації генетичний варіант відкине слабші варіанти і буде працювати з наборами, які показали найкращі результати на попередній ітерації.

### Висновки до розділу

Виконано огляд існуючих методів моделювання і прогнозування стаціонарних та нестаціонарних процесів та проведено аналіз їх переваг та недоліків. Додатково проведено огляд методики, що використовується для побудови моделей кандидатів, їх оцінки, для визначення тієї, що найкраще описує модельований процес, та подальшого використання для побудови прогнозів.

Також у розділі було розглянуто поняття штучної нейронної мережі, та базові поняття що з нею пов'язані. Було описано декілька видів нейронних мереж, що можуть використовуватися для прогнозування обсягів продажів на підприємстві. Було розглянуто їх структуру та принципи за якими відбираються параметри.

### 3 ПОБУДОВА МОДЕЛЕЙ ТА РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

В даному розділі описано структуру, результати перевірки та відбору моделей продажів підприємства. Розглянуто структуру побудованої СППР та наведено переваги використаних технологій. Також проведено огляд даних, що використовувались для тренування моделей, та на яких базувалась СППР.

#### 3.1 Огляд, аналіз та попередня обробка використаних даних

Для навчання моделей процесу продажів було використано дані відносно денних продажів мережі магазинів компанії «Rossmann». Дана компанія спеціалізується на продажі косметичних товарів, засобів догляду за тілом та побутової хімії. У даній сфері компанія є однією з найбільших в Європі, та становить мережу з понад 3000 магазинів.

Наданий набір даних складається з основного масиву даних, та додаткової інформації. Основний масив містить 1017211 запис, що відповідає денним продажам з 1115 магазинів в період з 01.01.2013 по 31.07.2015. Кожен запис містить наступну інформацію: Номер крамниці, день тижня, дата, обсяг продажів, к-сть відвідувачів, чи була відкрита крамниця, наявність розпродажі, чи був день національним вихідним або святом, чи були у цей день шкільні канікули.

Додатковий набір інформації містить дані відносно характеристик окремих магазинів. В даному наборі співставленні номери крамниць з такими параметрами:

приналежність до одного з 4х типів магазинів, оцінка асортименту крамниці від 1 до 3х, відстань до найближчого конкуруючого магазину, місяць та рік з

якого оперую конкурентна крамниця, участь у регулярних акціях, місяць та рік з яких діють такі акції та період у якому вони діють.

Для подальшої роботи з набором даних, основний та додатковий масив даних було поєднано у один набір, деякі дані було перетворено для полегшення роботи з ними та надання більшої інформативності. Так, наприклад, інформацію про регулярні акції було приведено до логічної змінної, що вказує на проведення акції у вказаний день. Данні про конкурентів було приведено до числового значення, що вказує кількість місяців оперування конкуренту. Типи крамниць та асортименти було приведено до числового еквіваленту. Через високу кореляцію з продажами, а також необхідність прогнозування, данні про кількість покупців було замінено на середню к-сть покупців у крамниці у конкретний день тижня.

Також, має сенс позбутися від усіх записів що відповідають дням коли магазин було зачинено, адже обсяг продажів завжди дорівнюватиме нулю. Якщо ж він не нульовий, то можна вважати такі дані помилковими, або виключними, так могли мати місце особливі обставини. Так або інакше, такі данні не несуть корисної інформації для навчання моделей.

Таким чином, дані було підготовлено для використання при навчанні моделей, та прогнозуванні. Було виділено єдину прогнозовану змінну, обсяг продажів, в той час як усі інші дані було зведено до загальної, або сезонної інформації, яку можна отримати без додаткових зусиль. Приклад отриманого набору даних наведено в додатку А.

## 3.2 Побудова та оцінка моделей

### 3.2.1 Моделі авторегресії

Для подальшого використання у СППР було побудовано та порівняно декілька моделей часових рядів. Для початку розглянемо моделі авторегресії.

Усі моделі було побудовано за допомогою мови програмування Python, та модулю Statmodels, який відповідає за спрощення процесу оцінку та перевірок статистичних моделей.

Варто зазначити, що для використання автокореляції, набір даних необхідно буде додатково розділити на часові ряди, що відповідають окремим крамницям, адже немає сенсу будувати автокореляційну модель з обсягів продажу різних крамниць так як вони, фактично, є різними показниками. Втім, через ідентичність процесу, можна вважати що модель побудована для однієї крамниці, буде ефективною і для інших.

Також, через простоту моделей, замість прогнозування рівнів продажі на пряму, можна побудувати модель, що прогнозуватиме кількість відвідувачів, так як автокореляція цього показника більша за автокореляцію обсягу продажів.

Спершу, необхідно визначити ступені автокореляції та ковзного середнього у моделях. Для цього розрахуємо АКФ та ЧАКФ (таблиця 3.1)

Таблиця 3.1 – Значення АКФ та ЧАКФ для продажів крамниці 1

Лаг	1	2	3	4	5	6	7	8	9	10	11	12
АКФ	0,219	0,193	-0,087	0,034	0,131	0,127	0,079	-0,049	0,126	0,008	0,191	0,187
ЧАКФ	0,219	0,153	-0,078	0,021	0,147	0,072	-0,005	-0,096	0,169	-0,030	0,127	0,147

Найбільших значень АКФ та ЧАКФ досягають при лагах 1 та 2. Втім, для в якості параметра автокореляції варто перевірити значення 6. Так як усі записи по продажам у неділю були видалені з набору даних, лаг 6 відповідатиме продажам у той самий день тижня неділю тому.

Також, варто перевірити результати при ковзному середньому з розміром вікна 5, яке буде відповідати середньому значенню продажів за тиждень.

Результати моделювання за допомогою АРКС моделей наступні:

Таблиця 3.2 – Моделі типу АРКС

Модель	АРКС(1,0)	АРКС(2,0)	АРКС(6,0)	АРКС(6,1)	АРКС(6,2)	АРКС(6,5)
AIC	10208	10191	10171	10173	10157	10134
R <sup>2</sup>	0,0479	0.0697	0.1002	0.1001	0.1197	0.1505
D.W.	2.0559	1.9608	1.9814	1.9814	1.9818	1.9767

Можна зробити висновок, що серед побудованих моделей, найкращою можна вважати АРКС(6,2). Графік та розподіл похибок для даної моделі можна побачити в додатку Б.

### 3.2.2 Моделі навчені нейронними мережами

Розглянемо побудовані нейронні мережі. В якості кандидатів були відібрані НМ перцептрону, багатошарового перцептрону та мережа РБФ. Усі НМ були побудовані за допомогою мови програмування Python, та бібліотек TensorFlow та Keras.

Tensorflow – це бібліотека машинного навчання, з фокусом на тренування та використання глибинних нейроммереж. Вона була розроблена та викладена у доступ з відкритим кодом компанією Google. Бібліотека Keras також є у відкритому доступі. Вона, фактично, є обгорткою над Tensorflow, що мінімізує зусилля необхідні для побудови нейроммереж різної складності.

Додатковою перевагою використання цих бібліотек в даній роботі стала наявність у відкритому доступі обгортки для них. Вчасності, обгортки для використання з мовою програмування C# та фреймворку .Net Core 5.0, на якій базується побудована СППР. Це означає, що попри те що аналіз та обробка даних, а також побудова експериментальних моделей, проводилась за допомогою мови Python, ми маємо можливість зберегти моделі та використати їх у роботі СППР що базується на технології .Net Core.



Порівняння побудованих мереж буде використовуватися за допомогою метрик середньоквадратичної похибки (MSE), її (RMSE) квадрату та квадрату відносної середньоквадратичної похибки (RMSPE). В якості функції активації буде використовуватися функція ReLU, а в якості методу оптимізації Адам.

Для початку, розглянемо просту мережу одношарового перцептрону. Одношаровий перцептрон містить, як випливає з назви, один прихований шар. Було вирішено провести навчання мереж згрупувавши дані на набори з 512 записів. Кількість нейронів у шарі відповідно обрана 512.

За основним прихованим шаром було розміщено шар допоміжний шар вибування (dropout). Цей шар використовується як додаткова міра запобігання перенавчання мережі. Основним параметром цього шару є його рівень дії rate, який приймає значення від 0 до 1. Цей рівень встановлює ймовірність кожного виходу мережі бути анульованим, в той час як усі інші виходи будуть помножені на значення  $1/(1 - \text{rate})$ , щоб зберегти загальну суму виходу відносно незмінною.

Наступною мережею кандидатом стала мережа Радіально Базисних функцій. Дана мережа так само складається з одного прихованого шару РБФ, та шару вибування що розміщується за ним. Центри радіально базисних функцій було обрано випадковим чином. Параметр  $\beta$  обрано вручну, зі значенням два.

Останньою мережею Став багатошаровий перцептрон. Було вирішено побудувати мережу з трьома прихованими шарами, з шарами вибування після кожного. Кожен шар був налаштовано ідентично до шару першої моделі.

Кількість ітерацій навчання усіх мереж було встановлено у 500 циклів, втім навчання зупиниться якщо значення функції втрат для тестових даних не буде покращуватися на протязі 10 ітерацій.

Після побудови моделей на нормалізованих даних, отримано наступні результати для тестових даних:

Таблиця 3.3 – Порівняння метрик НМ

Мережа	К-сть ітерацій	MSE	RMSE	RMSPE
Одношаровий	500	0,0969	0,3105	1,4850
РБФ	44	0,2001	0,4462	2,1932
Багатошаровий	253	0,0848	0,2903	1,4299

Як можна побачити, моделі побудовані нейромережами є більш точнішими, що пояснюється як і наявністю процесу навчання, так і використанням додаткових змінних. Також, очевидним фаворитом стала модель багатошарового перцептрону. Це легко пояснюється більш складною структурою мереж та самим процесом побудови моделей.

Втім, як і варто було очікувати, більш складна структура мережі значно збільшила час та обчислювальні ресурси що були необхідні для отримання моделей.

Модель багатошарового перцептрону було збережено для подальшого використання у СППР.

Результати прогнозування за допомогою багатошарового перцептрону можна побачити на таблиці 3.4

Таблиця 3.4 – Приклад прогнозування за допомогою багатошарового перцептрону

Реальне	7699	5426	8539	6521	5321	7388	4071	9151
Прогнозоване	8764,711	5353,1416	8142,529	6108,25	5796,9956	7837,66	4928,6973	8762,074

Не зважаючи на наявну похибку, результати прогнозування можна вважати прийнятними, так як вони в достатній мірі описують ситуацію відносно майбутніх продажів підприємства. Також, саму похибку можна пояснити відносно простою структурою самої нейромережі, та обмеженістю даних які вона використовує.

### 3.3 Загальний огляд реалізованої СППР

Створена в рамках даної роботи СППР складається з двох частин, клієнтська частина, фронтенд, та серверна, бекенд. Бекенд було побудовано за допомогою мови програмування C# на базі технології .Net Core, яка представляє собою фреймворк для реалізації веб додатків та API (Application Programming Interface – набір підпрограм який використовується для поєднання інших сервісів).

Фреймворк .Net Core було розроблено компанією Microsoft, як кроссплатформенний та модульний аналог їх популярного фреймворку .Net Framework. Данна технологія значно спрощує та при цьому надає більшу гнучкість при розробці додатків. Вона містить у собі усі необхідні технології для швидкої реалізації простого API в якості бекенду СППР.

Для створення фронтенду, було задіяно скриптову мову JavaScript та технології NodeJS, необхідну для створення середовища розробки, та бібліотеки ReactJS, Redux, Bootstrap та інших, допоміжних бібліотек.

ReactJS – це бібліотека, що дозволяє створювати гнучкі, динамічні веб сторінки. Вона побудована на принципі модальності сторінок, та дозволяє легко повторно використовувати вже існуючі елементи веб сторінки на інших. В цілому, найчастіше її використання – розробка SPA (Single Page Application – додаток з єдиною сторінкою ). Фактично, створений за таким принципом веб інтерфейс міститиме єдину веб сторінку, під час роботи якої будуть змінюватися лише певні її елементи.

В тандемі з ReactJS, для веб розробки часто використовують бібліотеку Redux. Її основна функція – забезпечити веб-інтерфейс єдиним сховищем станів усього додатку та окремих модулів, а також забезпечити додаток сховищем для отриманих з серверу даних.

В цілому, СППР було реалізовано за структурою популярного шаблону MVC –Модель-Вид-Контролер. Загальну структуру та взаємодію між частинами програми зазначено на рисунку 3.1.

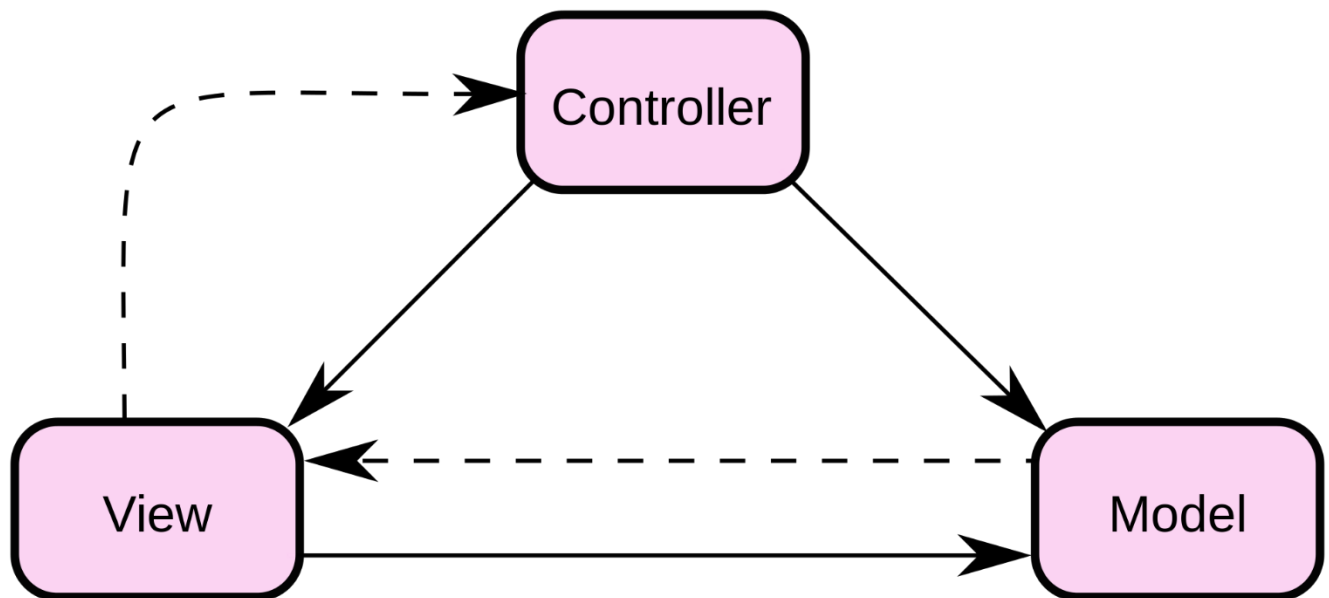


Рисунок 3.1 – Схема структури MVC шаблону

Модуль «Модель» є центральним компонентом шаблону і відповідає моделі даних, яку використовує додаток, а також описує логіку та правила взаємодії з даними, і, фактично, є посередником між бекендом додатку та базою даних. Вона також містить у собі увесь функціонал для обробки даних, наприклад, у випадку з СППР, модель нейромереж та функціонал побудови прогнозу.

Модуль «Контролер» є посередником між видом та моделю. У розробленій СППР, бекенд реалізовано як API, тому він покриває лише функціонали моделі та контролеру. Задача контролеру, в такому випадку – отримання та обробка HTTP запитів від користувацького інтерфейсу, передача інструкцій моделі, якщо це необхідно, та надання відповіді клієнту.

Модуль «Вид», у випадку даної СППР, є відокремленим від контролера та моделі, та використовує інші технології. Проте повністю їх розділити

неможливо, адже без серверної частини клієнт не матиме доступу до даних та функціоналу СППР, а без клієнтської частини буде відсутній користувацький інтерфейс. З цього і випливає основна задача даного модулю – надання користувачу можливості оглядати, та маніпулювати даними системи.

Виходячи з використаних технологій можна сформулювати технічні вимоги для роботи системи.

Для роботи клієнтської частини необхідний браузер, що підтримує мову JavaScript, тобто, майже будь-який сучасний веб-браузер, незалежно від пристрою та операційної системи пристрою.

Для роботи серверної частини буде необхідним пристрій з ОС Windows (версія 7 та пізніше), Linux або MacOS. Через використання бібліотеки TensorFlow 2, процесор пристрою повинен мати розрядність 64біт, та, принаймні, 4ГБ оперативної пам'яті.

### 3.4 Огляд інтерфейсу користувача

Інтерфейс користувача реалізовано у широко використаному форматі дашборду. В горі вікна присутня панель-заголовок. Вона несе декоративний характер, та містить лише робочу назву проекту. В основному, на таких панелях можуть бути розміщені пошукові рядки та шлях до активної сторінки (breadcrumbs), що може використовуватись для швидкого переходу між попередніми вікнами, та посилання на елементи програми, які не пов'язані з її основною функціональністю, і надають лише допоміжну функціональність, наприклад сторінка користувача, налаштувань та інше.

Головне вікно додатку реалізує втілює принцип SPA (single-page application), і відображає модуль залежно від обраного значення на панелі навігації, яка розміщена зліва. Кожна модель відображення на головному вікні

прив’язана до конкретної URL адреси, аналогічно до звичайних веб сторінок, втім, при переході між адресами веб-браузер не завантажує сторінку цілком, а тільки оновлює змінюваний елемент. Така архітектура надає переваги як при використанні додатку, так і при його розробці, адже усі статичні елементи інтерфейсу, панель-заголовок та навігаційна панель, розроблюються у єдиному екземплярі, що значно спрощує процес внесення змін.

Фактично, усі статичні та динамічні елементи інтерфейсу, панелі та вікна, виконано у форматі окремих компонент. Задачу їх компонування та компілювання у відповідні статичні HTML сторінки виконує бібліотека ReactJS.

Зв’язок та обмін даними між клієнтською частинами відбувається за допомогою REST-запитів. Було прийняте рішення позбавити клієнтську частину можливостей видалення та внесення змін до інформації, що зберігаються у базі даних, тому система використовує лише GET-запити, які необхідні для отримання інформації, що буде відображена у системі, або для отримання прогнозів, відповідно до вводу користувача.

В якості сторінки за замовчання, було розроблено компонент “HomePageComponent”, який представляє собою домашню сторінку системи. Інтерфейс домашньої сторінки зображено на рисунку 3.2.

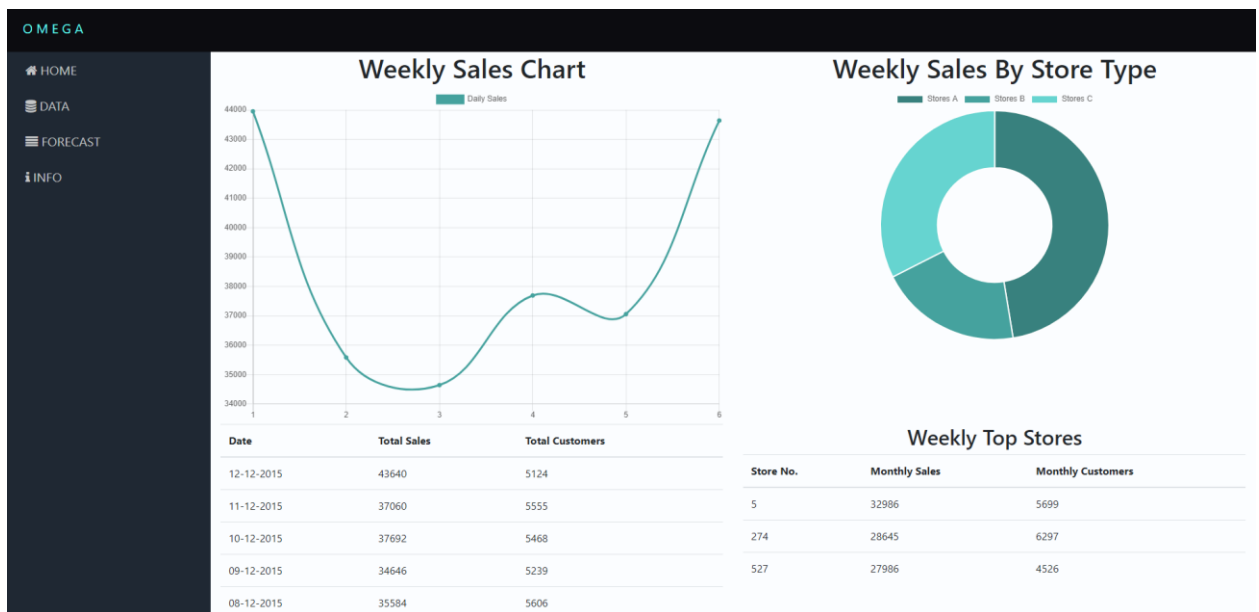


Рисунок 3.2 – Домашня сторінка додатку

Домашня сторінка несе функцію відображення поверхневого огляду даних, що зберігається у системі. Конкретніше, на ній присутні чотири елементи:

- тижневий графік загальних продажів;
- відповідність загальної кількості тижневих продажів до категорії магазину;
- таблиця загальних продажів за 6 останніх днів;
- таблиця що відображає топ-3 магазини за обсягом продажів.

В цілях оптимізації роботи користувацького інтерфейсу, з серверної частини передаються уже сформовані, та готові до використання дані. Клієнтській частині необхідно лише співставити отримані данні з елементом, який їх відобразить.

Посилання “DATA” на навігаційній панелі змінює головне вікно на панель детального огляду даних, реалізовано у компоненті “DataviewComponent”. Інтерфейс сторінки зображено на рисунку 3.3.

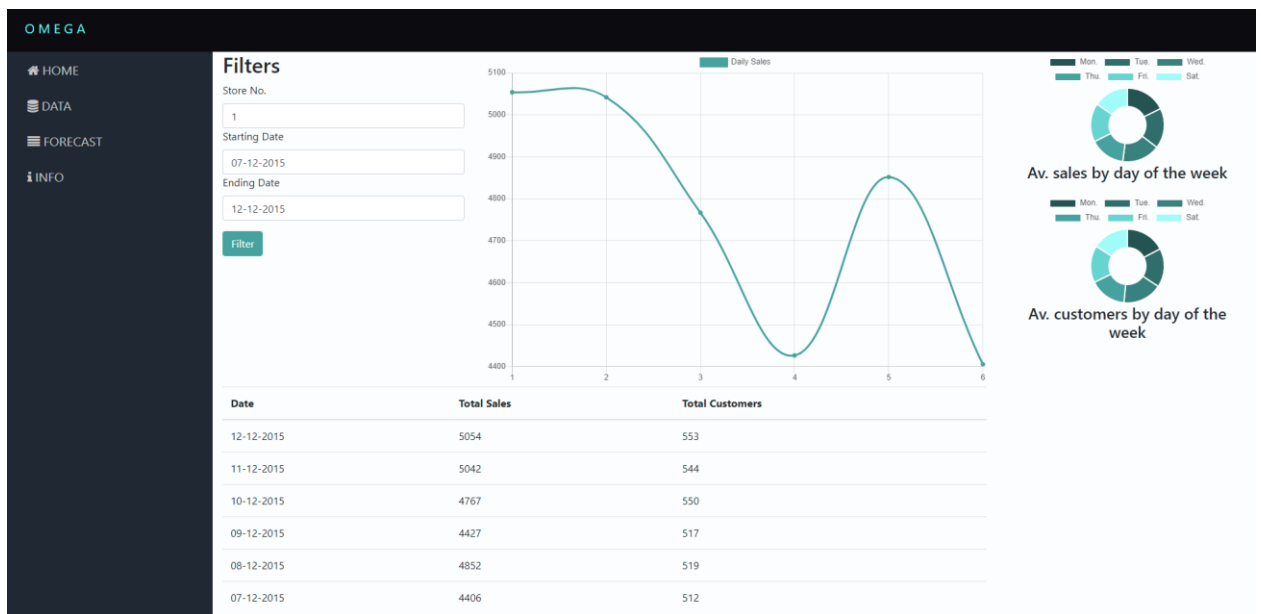


Рисунок 3.3 – Сторінка огляду даних

На даній сторінці представлено детальний огляд даних по продажам, залежно від вказаного фільтру. Фільтрація даних потребує три параметри: номер крамниці, дату початку та завершення періоду. Після натискання на кнопку “Filter” система відобразить отримані дані у таблиці внизу сторінки, побудує графік на базі цих даних, а також надасть інформацію відносно середньої кількості продажів та покупців, залежно від дня тижня.

Остання реалізована сторінка доступна через кнопку “Forecast” на навігаційній панелі, і реалізована у компоненті “ForecastComponent”. Відповідно до назви у ній і реалізовано функціонал прогнозування даних. Інтерфейс зображено на рисунку 3.4

The screenshot shows the OMEGA dashboard with the 'Forecast' component. It includes input fields for 'Store No.' (5) and 'Date' (01.12.2015). Below these are three checkboxes: 'State Holiday' (unchecked), 'School Holiday' (checked), and 'Promotion Is Active' (unchecked). A 'Get Sales Forecast' button is present. Below the button is a 'Forecasted Sales' section displaying the value '7123,12'.



### Рисунок 3.4 – Сторінка прогнозування

Дане вікно є досить мінімалістичним, що значно спрощує процес прогнозування продажів. Незважаючи на те, що модель прогнозування використовує більше 30 вхідних параметрів, користувачу достатньо ввести лише 5. Цими параметрами є номер магазину, дата прогнозування, та 3 логічні признаки що відповідають дії акції, загального вихідного дня чи свята, та шкільних канікул. Усі інші показники будуть автоматично отримані з бази даних.

Потенційно, для прогнозування продажів на одному магазині в конкретний день, вхідні дані користувача можна звести власне до номеру магазину та дати. Втім, такий підхід вимагав би наявності у системі інформації про свята, вихідні дні та шкільні канікули, що є не дуже доцільним в рамках даної дисертації.

Іншим напрямком покращення системи є розробка функціоналу прогнозування продажів для кількох магазинів, одночасне прогнозування на тривалий період часу, та повна автоматизація побудови прогнозів.

Нейронна мережа, що використовується для прогнозування, цілком підтримує можливість прогнозу для ряду даних, втім розробка такого функціоналу потребувала б додаткового часу, зусиль та використання більшої кількості бібліотек, як на серверній стороні, так і у клієнті. Головною проблемою є додавання таблиць що дозволяють редагування даних. Робота з такими таблицями вимагає складніший функціонал збереження та оновлення відображуваних даних, а також їх передачу та обробку на стороні серверу.

Повна автоматизація побудови прогнозів також не є доцільною в рамках даної роботи. Це пов'язане з двома проблемами. Перша проблема це необхідність створення системи користувацьких налаштувань для прогнозів. Такі налаштування будуть диктувати на базі яких даних та за який період будується прогноз.

Інша проблема, що не дозволяє автоматизувати прогнозування в рамках даної роботи полягає у її експериментальній сутності. Система не перебуває у постійному використанні, у неї не потрапляють нові дані, а дані що занесені у систему є застарілими. Фактично це означає, що у системі відсутні будь-які тригери, які б могли викликати побудову прогнозу, та зберегти його для подальшого використання. Також у системі відсутня інформація про зовнішні показники, наприклад, зазначені раніше, канікули та свята.

Автоматизація прогнозу матиме сенс у тому випадку, коли система буде інтегрована з іншою, живою ERP (Enterprise Resource Planning) системою на реальному підприємстві, або ж повне перетворення розробленої СППР у неї. Таке рішення вирішило б поставлені вище проблеми.

### Висновки до розділу

В даному розділі виконано огляд набору даних про обсяг продажів, та процес попередньої обробки цих даних для використання при побудові моделей часових рядів та навчанні нейромереж. Описано процес та технічні деталі побудови цих моделей в якості кандидатів, та їх подальший відбір для використанні в СППР.

Для використанні в СППР відібрана штучна нейронна мережа багатошарового перцептрону. Також, в розділі описана архітектура спроектованої СППР, як клієнтської, так і серверної частини. Детально розглянуто інтерфейс користувача.

Розроблена система відповідає вимогам СППР, а саме: має зручний користувацький інтерфейс, надає інструменти для огляду та аналізу даних, а також можливість будування прогнозів. Перевагою розробленого продукту є його простота та зручність використання, інтуїтивний інтерфейс, а також відносна точність роботи.



## 4 СТАРТАП АНАЛІЗ ПРОЕКТУ

### 4.1 Інформаційна карта проекту

Ідея та реалізація стартап проекту зображена у таблицях 4.1 – 4.19

Таблиця 4.1 – інформаційна карта проекту

1. Назва проекту	Система підтримки прийняття рішень для прогнозування рівнів продажу на підприємстві
2. Автори проекту	Худецький Михайло
3. Коротка анотація (не більше 1/3 сторінки)	Дана СППР призначена для прогнозування обсягів продажу на підприємстві. Система зберігає, та надає можливість переглянути, існуючі дані стосовно продажів в мережі магазинів. На основі наявних історичних даних, система надає прогноз на майбутні періоди.
4. Термін реалізації проекту	6 місяців
5. Необхідні ресурси	Обладнання – комп'ютери, принтер, сканер, ноутбуки, мобільні телефони. Програмне забезпечення, операційна система, ліцензії на комерційну розробку ПЗ. Електрика, газ, водопостачання, Інтернет. Фінансові ресурси – заробітна плата працівникам на 6 місяців роботи, гроші на оплату комунальних послуг,

Продовження таблиці 4.1

	оренди, реклами тощо. Приміщення з усіма необхідними комунікаціями.
6. Опис проблеми, яку вирішує проект	Дана комплексна система дозволяє зберігати інформацію щодо продажів в мережі магазинів, а також надає прогноз майбутніх продажів для окремих магазинів, в заданий період часу.
7. Головні цілі та завдання проекту	Розробка комплексної системи, робота із реальними даними та створення комерційно успішного продукту.
8. Очікувані результати	Система дозволяє побудувати якомога точний прогноз для мережі магазинів або окремих крамниць, в заданий період часу, та наданою інформацією про зовнішні фактори.

## 4.2 Технологічний аудит ідеї проекту

Таблиця 4.2 - Опис ідеї стартап-проекту

<i>Зміст ідеї</i>	<i>Напрямки застосування</i>	<i>Вигоди для користувача</i>
Дана СППР призначена для прогнозування обсягів продажу на підприємстві. Система зберігає, та надає можливість переглянути, існуючі дані стосовно продажів в мережі магазинів. На основі наявних історичних даних та побудованої нейронної мережі, багат шарового перцептронну, система надає прогноз на майбутні періоди.	1.Робота з існуючими даними.	Система дозволяє замовникам спростити доступ
	2.Побудова прогнозів відносно обсягів продажів.	Система дозволяє замовникам зменшити витрати часу та людських ресурсів на створення прогнозу.

Таблиця 4.3 - Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ n/n	Техніко- економічні характеристики ідеї	(потенційні) товари/концепції конкурентів			W	N	S
		Власний проект	Pipedrive	Zoho CRM			
1.	Точність прогнозування	Застосування моделі на базі нейромережі	Залежить від вхідних даних	Залежить від вхідних даних			+
2.	Ризики невірного прогнозу	Існують, через обмежену кількість вхідних параметрів.	Залежить від вхідних даних	Залежить від вхідних даних		+	
3.	Доступність зручність	Графічний інтерфейс на основі ReactJS	Власний інтерфейс	Власний інтерфейс		+	

Таблиця 4.4 - Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Створення Системи підтримки	Використання фреймворку .Net Core	Наявна	Доступні
2	прийняття рішень	Використання фреймворку .Net Framework	Наявна	Доступні
3		Використання фреймворку Django	Наявна	Доступні
Обрана технологія реалізації ідеї проекту: .Net Core				



### 4.3 Аналіз ринкових можливостей запуску стартап-проекту

Таблиця 4.5 - Попередня характеристика потенційного ринку стартап-проекту

<i>№ n/n</i>	<i>Показники стану ринку (найменування)</i>	<i>Характеристика</i>
1	Кількість головних гравців, од	2
2	Загальний обсяг продаж, грн/ум.од	1000
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	14%

Таблиця 4.6 - Характеристика потенційних клієнтів стартап-проекту

<i>№ n/n</i>	<i>Потреба, що формує ринок</i>	<i>Цільова аудиторія (цільові сегменти ринку)</i>	<i>Відмінності у поведінці різних потенційних цільових груп клієнтів</i>	<i>Вимоги споживачів до товару</i>
1	Статистичні параметри роботи мережі магазинів	Аналітики, аналітичні відділи	Низька ціна, велика кількість статистичних даних.	Простота використання.
2	Створення якісного прогнозу продажів при обмежених даних.	Малі та середні виробничі або комерційні підприємства.	Цікавить простота у використанні, низька ціна клієнтська підтримка	Низька ціна, репутація
3	Створення точного довгострокового прогнозу продажів.	Великі підприємства	Цікавить передусім точність довгострокових прогнозів, клієнтська підтримка, інтеграція до власних потреб	Висока якість, бренд, ім'я на ринку, успішний досвід

Таблиця 4.7 - Фактори загроз

<i>№ п/п</i>	<i>Фактор</i>	<i>Зміст загрози</i>	<i>Можлива реакція компанії</i>
1	Конкуренція	Високий рівень конкуренції	Розробити маркетингову компанію з акцентом на персоналізацію продукту
	Збут	Ускладнення збуту через цінову політику конкурентів	Розміщення додаткової реклами

Таблиця 4.8 - Фактори можливостей

<i>№ п/п</i>	<i>Фактор</i>	<i>Зміст можливості</i>	<i>Можлива реакція компанії</i>
1	Гнучкі ціни	Зменшення ціни товару задля збільшення попиту	Введення власних гнучких цін
2	Диференціація витрат	Зменшення витрат за рахунок їх перерозподілу	Зменшення витрат на додаткові, непрофільні задачі.

Таблиця 4.9 - Ступеневий аналіз конкуренції на ринку

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)</i>
1. Тип конкуренції - Досконала конкуренція	Багато систем/команд аналітиків	Розробити впізнаваний продукт, якість, що вирізнятиме нас від конкурентів
2. За рівнем конкурентної боротьби: міжнародний	На ринку присутні системи, розроблені в інших країнах.	Розширення аудиторії, розширення списку мов, яві підтримуються системою
3. За галузевою ознакою - міжгалузева	Робота із різногалузевими підприємствами.	Покращення можливостей персоналізації системи до вимог клієнта.
4. Конкуренція за видами товарів: товарно-родова	Конкуренція між прогнозами інших інформаційних систем та команд аналітиків.	Збільшення функціоналу системи
5. За характером конкурентних переваг: Нецінова	Різні способи прогнозування дають різну точність	Розробка кращих(точніших) алгоритмів
6. За інтенсивністю: марочна	Впізнаваний бренд надає великих переваг	Велику увагу приділити розвитку бренду

Таблиця 4.10 - Аналіз конкуренції в галузі за М. Портером

<i>Складові аналізу</i>	<i>Прямі конкуренти в галузі</i>	<i>Потенційні конкуренти</i>	<i>Постачальники</i>	<i>Клієнти</i>	<i>Товари-замінники</i>
	Інші СППР	Гнучкі ціни, розмір капіталовкладень	Змінні витрати постачальників, диференціація витрат	Контроль якості, система інформації	Ціна, лояльність споживачів
<i>Висновки</i>	Інтенсивна конкуренція	Є як можливості входження на ринок, так і нові потенційні конкуренти	Постачальники не диктують умови роботи на ринку	Клієнти не диктують умови роботи на ринку	Обмеження для роботи на ринку через товари замінники

Таблиця 4.11 - Обґрунтування факторів конкурентоспроможності

<i>№ n/n</i>	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Простота використання	Система будує точні прогнози при мінімальному вводі користувача
2	Якість	Висока якість прогнозу, велика кількість допоміжних статистичних даних
3	Обслуговування	Робота з клієнтами – передусім малим та середнім бізнесом

Таблиця 4.12 - Порівняльний аналіз сильних та слабких сторін системи

<i>№ n/n</i>	<i>Фактор конкурентоспроможності</i>	<i>Бали 1-20</i>	<i>Рейтинг товарів-конкурентів</i>						
			-3	-2	-1	0	+1	+2	+3
1	Простота використання	15		2			3		1
2	Якість	20	3	2	1		2		
3	Обслуговування	5					3	1	2

Таблиця 4.13 - SWOT- аналіз стартап-проекту

<p>Сильні сторони:</p> <p>Висока якість прогнозу</p> <p>Інтерфейс користувача</p> <p>Простота використання</p>	<p>Слабкі сторони:</p> <p>Відсутність клієнтської бази</p> <p>Відносно мала функціональність</p>
<p>Можливості:</p> <p>Попит</p> <p>Зміна рівня доходів підприємств</p> <p>Вдосконалення системи</p>	<p>Загрози:</p> <p>Конкуренція</p> <p>Збут</p>

Таблиця 4.14 - Альтернативи ринкового впровадження стартап-проекту

<i>№ п/п</i>	<i>Альтернатива (орієнтовний комплекс заходів) ринкової поведінки</i>	<i>Ймовірність отримання ресурсів</i>	<i>Строки реалізації</i>
1	Швидкий вихід на ринок із «сирим» продуктом, можливі проблеми із точністю прогнозу та попитом	30%	3 місяці
2	Поступовий вихід з готовим, відлагодженим продуктом. Висока якість та конкурентоспроможна ціна.	70%	6 місяців

#### 4.4 Розроблення ринкової стратегії проекту

Таблиця 4.15 - Вибір цільових груп потенційних споживачів

<i>№ п/п</i>	<i>Опис профілю цільової групи потенційних клієнтів</i>	<i>Готовність споживачів сприйняти продукт</i>	<i>Орієнтовний попит в межах цільової групи (сегменту)</i>	<i>Інтенсивність конкуренції в сегменті</i>	<i>Простота входу у сегмент</i>
1	Окремі аналітики та аналітичні відділи невеликих компаній	Висока	2%	Висока	Середня
2	Малі та середні підприємства	Середня	1%	Висока	Середня
3	Великі підприємства	Низька	1%	Середня	Висока
Які цільові групи обрано: 1,2					



Таблиця 4.16 - Визначення базової стратегії розвитку

<i>№ п/п</i>	<i>Обрана альтернатива розвитку проекту</i>	<i>Стратегія охоплення ринку</i>	<i>Ключові конкурентоспроможні і позиції відповідно до обраної альтернативи</i>	<i>Базова стратегія розвитку *</i>
1	2	Стратегія диференційованого маркетингу	Високі можливості персоналізації, багатогалузовість, висока якість, ціна.	Стратегія лідерства по витратах

Таблиця 4.17 - Визначення базової стратегії конкурентної поведінки

<i>№ п/п</i>	<i>Чи є проект «першопрохідцем» на ринку?</i>	<i>Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?</i>	<i>Чи буде компанія копіювати основні характеристики товару конкурента, і які?</i>	<i>Стратегія конкурентної поведінки*</i>
	Ні	Так	Ні	Стратегія виклику лідера

Таблиця 4.18 - Визначення стратегії позиціонування

<i>№ п/п</i>	<i>Вимоги до товару цільової аудиторії</i>	<i>Базова стратегія розвитку</i>	<i>Ключові конкурентоспромо жні позиції власного стартап-проекту</i>	<i>Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)</i>
1	Якість, точність, простота у використа нні	Стратегія лідерства по витратах	Якість прогнозу, велика кількість статистичної інформації	По іміджу Позиціонування на низькій ціні позиціонування за сферою застосування

#### 4.5 Розроблення маркетингової програми стартап-проекту

Таблиця 4.19 - Визначення ключових переваг концепції потенційного товару

<i>№ п/п</i>	<i>Потреба</i>	<i>Вигода, яку пропонує товар</i>	<i>Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)</i>
1	Статистичні параметри роботи мережі магазинів	Базова статистика та довгострокові прогнози	Дана статистика має бути максимально розгорнутою та зрозумілою.
2	Створення якісного прогнозу продажів при обмежених даних.	Побудова прогнозів при малому обсязі даних	Можливість інтеграції СППР у підприємствах, які тільки недавно розпочали дію.
3	Створення точного довгострокового прогнозу продажів.	Точний прогноз продажів – основа функціонування будь-якого підприємства.	Розробка коротко- та довгострокових прогнозів, використання нейромережі для покращення точності прогнозу

#### 4.6 Висновки до розділу

Було побудовано стартап проект системи підтримки прийняття рішень, яка дозволяє будувати прогноз обсягів продажів для мережі магазинів. Тривалість проекту 6 місяців. Були відокремлені основні чинники для фіксації ризиків та факторів, які зможуть суттєво вплинути на даний проект.

## ВИСНОВКИ

В даній роботі виконано огляд методів та методик що використовуються для прогнозування рівнів продажу на підприємствах. Було визначено поняття прогнозу продажів, важливість наявності такого прогнозу для планування роботи підприємства, торгівельного або виробничого. Було детально розглянуто обидва типи прогнозів продажів, якісні та кількісні. Визначено порядок їх формування, переваги, недоліки, та ситуації у яких варто скористатися конкретними методами.

У другій частині роботи розглянуто та описано моделі часових рядів, що можуть використовуватися для прогнозування продажів підприємства у разі використання кількісних методів прогнозування. Описано методику підготовки та відбору моделей, та їх використання в задачах прогнозування. Також розглянуто деякі штучні нейронні мережі, багатошаровий перцептрон та мережі РБФ. Було описано в загалом які задачі можуть бути виконані за допомогою даних мереж, особливості їх структури, недоліки та переваги.

У третій частині описано структуру, параметри та характеристики побудованих моделей та нейронних мереж. Проведено огляд та аналіз даних, на яких навчались моделі. Для використання у СППР було відібрано нейронну мережу багатошарового перцептрону. Описано структуру розробленої СППР, та розглянуто переваги технологій, які використовувались для її реалізації.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Sales Forecasting. Електронний ресурс.  
URL: <https://www.skyword.com/marketing-dictionary/sales-forecasting/>
2. Sales Forecasting: Meaning, Importance and Methods. Електронний ресурс.  
URL: <https://www.businessmanagementideas.com/sales/forecasting-sales/sales-forecasting-meaning-importance-and-methods/7122>
3. Mentzer J.T., Moon M.A. Sales forecasting management: A demand management approach. 2nd ed. London: Sage Publications Inc. 2005. 351 p.
4. Lancaster G. A., Lomas R. A. Forecasting for sales and materials management. London: Macmillan publishers LTD. 1985. 191 p.
5. Chambers J.C., Mullick S.K., Smith D.D. How to choose the right Forecasting Technique. *Harvard Business Review*. 1971. July 01. p. 25.
6. Бідюк П.І. Системний підхід до прогнозування на основі моделей часових рядів. *Системні дослідження та інформаційні технології*. 2003. № 3. С. 88 – 110.
7. Box G. E. P., Jenkins G.M. Reinsel G.C. Time Series Analysis: Forecasting and Control (Third ed.). Eaglewood Cliffs, NJ: Prentice-Hall. 1994. 598 p.
8. Poggio T., Girosi F. Networks for approximation and learning. *Proceedings of the IEEE*. 1990. Vol. 78, No. 9. P. 1481-1497.
9. Buhmann M. D. Radial Basis Functions: Theory and Implementations. Cambridge: Cambridge University Press. 2003. 258p.

## ДОДАТОК А

Приклад структури використаних даних після обробки

Таблиця А.1

	Sales	DayOfWeek	Open	Promo	SchoolHoliday	CompetitionDistance
0	5263.0	5	1.0	1	1	1270.0
1	6064.0	5	1.0	1	1	570.0
2	8314.0	5	1.0	1	1	14130.0
3	13995.0	5	1.0	1	1	620.0

Продовження таблиці А.1

CompetitionOpenSinceMonth	CompetitionOpenSinceYear	Promo2	Promo2SinceWeek	Promo2SinceYear
9.0	2008.0	0	0.0	0.0
11.0	2007.0	1	13.0	2010.0
12.0	2006.0	1	14.0	2011.0
9.0	2009.0	0	0.0	0.0

Продовження таблиці А.1

Wapp	Avg_Customers	Year	Month	Day	CompetitionOpen
1012.0	537.0	2015	7	31	82.0
1013.0	537.0	2015	7	31	92.0
1098.0	747.0	2015	7	31	103.0
870.0	1245.0	2015	7	31	70.0

Продовження таблиці А.1

PromoOpen	IsPromoMonth	Store	StateHoliday_0	StateHoliday_a	StateHoliday_b
0.0	0	1	1	0	0
64.5	1	2	1	0	0
52.25	1	3	1	0	0
0.0	0	4	1	0	0

Продовження таблиці А.1

StateHoliday_c	StoreType_a	StoreType_b	StoreType_c	StoreType_d
0	0	0	1	0
0	1	0	0	0
0	1	0	0	0
0	0	0	1	0

Продовження таблиці А.1

Assortment_a	Assortment_b	Assortment_c
1	0	0
1	0	0
1	0	0
0	0	1



**ДОДАТОК Б**

Графік та розподіл похибок для моделі АРКС(6,2).

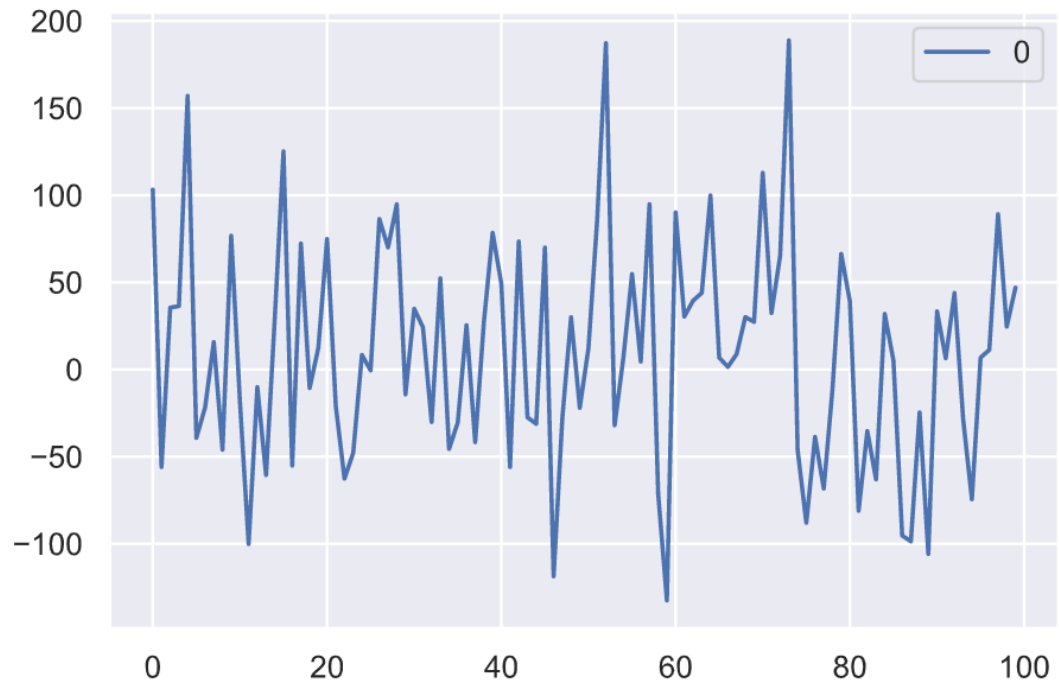


Рисунок Б.1

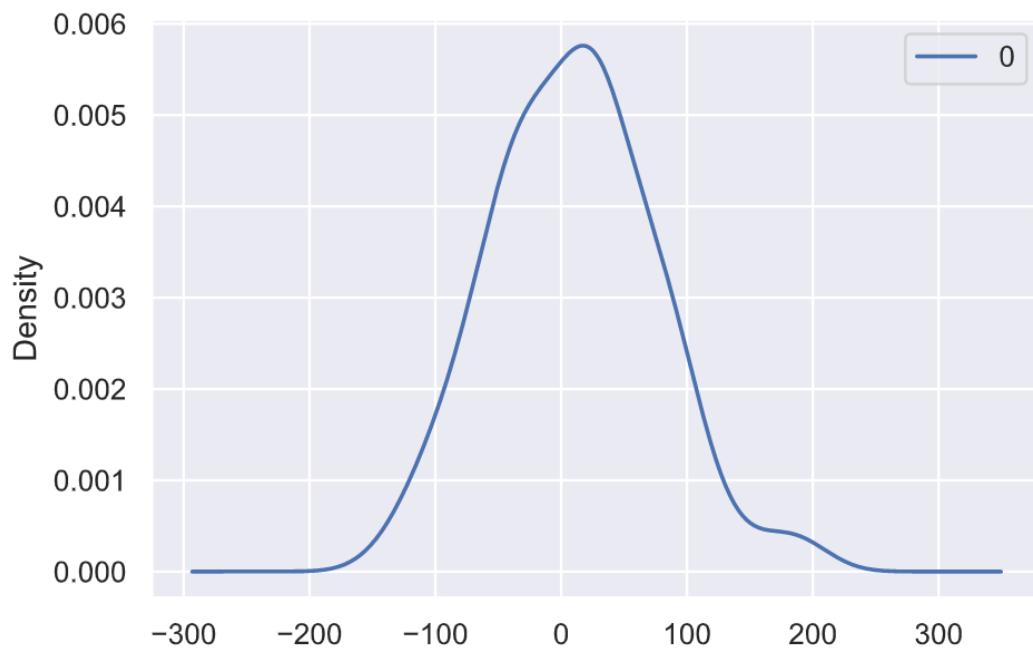


Рисунок Б.2

## ДОДАТОК В

### Лістинг коду для моделі

```
from statsmodels.tsa.arima_model import ARIMA
model = ARIMA(data_arima, order=(6,0,2))
model_fit = model.fit(dis=0)
print(model_fit.summary())
```

### Лістинг коду РБФ мережі

```
def create_model():
    model = Sequential()
    rbflayer = RBFLayer(10,
                        initializer=InitCentersRandom(X_train),
                        betas=2.0,
                        input_shape=(X_train.shape[1],))
    outputlayer = Dense(1, use_bias=False)
    model.add(rbflayer)
    model.add(outputlayer)
    model.compile(loss='mean_squared_error',
                  optimizer=RMSprop(),
                  metrics=[rmse, rmspe])
    return model
```

### Лістинг коду багатошарового перцептрону

```
def create_model():
    initializer = RandomNormal(mean=0.0, stddev=0.05, seed=None)
    model = Sequential()
    model.add(Dense(512, input_dim=X_train.shape[1], activation="relu", kernel_initializer=initializer))
    model.add(Dropout(0.4))
    model.add(Dense(512, input_dim=X_train.shape[1], activation="relu", kernel_initializer=initializer))
    model.add(Dropout(0.4))
    model.add(Dense(512, input_dim=X_train.shape[1], activation="relu", kernel_initializer=initializer))
    model.add(Dropout(0.4))
    model.add(Dense(1, activation="linear", kernel_initializer=initializer))
    adam = Adam(lr=1e-3, decay=1e-3)
    model.compile(loss="mean_squared_error", optimizer=adam)
    return model
```